

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA

SUBDIRECCIÓN DE ESTUDIOS DE POSGRADO



HIPERHEURÍSTICO MULTIOBJETIVO BASADO EN
BÚSQUEDA DE VECINDAD VARIABLE PARA PROBLEMAS DE
SELECCIÓN DE CARTERA DE PROYECTOS

POR

MADAY HERNÁNDEZ QUEVEDO

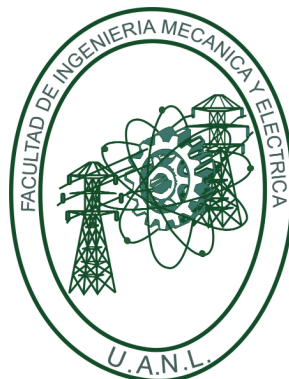
COMO REQUISITO PARCIAL PARA OBTENER EL GRADO DE
MAESTRÍA EN CIENCIAS DE LA INGENIERÍA CON ORIENTACIÓN
EN SISTEMAS

SEPTIEMBRE 2020

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA

SUBDIRECCIÓN DE ESTUDIOS DE POSGRADO



HIPERHEURÍSTICO MULTIOBJETIVO BASADO EN
BÚSQUEDA DE VECINDAD VARIABLE PARA PROBLEMAS DE
SELECCIÓN DE CARTERA DE PROYECTOS

POR

MADAY HERNÁNDEZ QUEVEDO

COMO REQUISITO PARCIAL PARA OBTENER EL GRADO DE
MAESTRÍA EN CIENCIAS DE LA INGENIERÍA CON ORIENTACIÓN
EN SISTEMAS

SEPTIEMBRE 2020



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

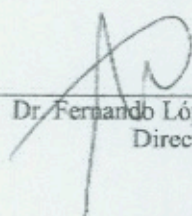


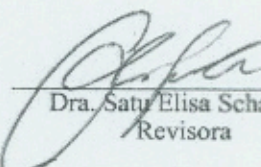
FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA

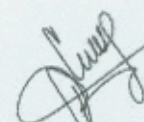
Universidad Autónoma de Nuevo León
Facultad de Ingeniería Mecánica y Eléctrica
Subdirección de Estudios de Posgrado

Los miembros del Comité de Tesis recomendamos que la Tesis "Hiperheurístico multiobjetivo basado en búsqueda de vecindad variable para problemas de selección de cartera de proyectos", realizada por el alumno Maday Hernández Quevedo, con número de matrícula 1985280, sea aceptada para su defensa como requisito para obtener el grado de Maestría en Ciencias de la Ingeniería con Orientación en Sistemas.

El Comité de Tesis


Dr. Fernando López Irarragorri
Director


Dra. Saty Elisa Schaeffer
Revisora


Dra. Laura Cruz Reyes
Revisora

Vo. Bo.


Dr. Simón Martínez Martínez
Subdirector de Estudios de Posgrado



063

San Nicolás de los Garza, Nuevo León, octubre de 2020



Ciudad Universitaria Pedro de Alba s/n, C.P. 66455, A.P. 076 Suc. "F"
San Nicolás de los Garza, Nuevo León, México. Tels: (81) 8332 0903 /
Comm: 8329 4020 / Fax: (81) 8332 0904

ÍNDICE GENERAL

1. Introducción	1
1.1. Justificación y motivación	2
1.2. Diseño metodológico	4
1.2.1. Problema	5
1.2.2. Objetivos e hipótesis	5
1.2.3. Novedad científica	5
1.2.4. Alcance y limitaciones	6
2. Marco teórico	7
2.1. Clasificación de los problemas de decisión	7
2.2. Proceso de apoyo a la toma de decisión multicriterio	9
2.3. Métodos toma de decisiones bajo múltiples criterios	11
2.4. Optimización multiobjetivo	15
2.5. Optimización lineal entera	18
2.5.1. Clasificación de problemas lineales enteros	19

2.6. Hiperheurísticos	20
2.7. Búsqueda de vecindad variable	22
2.8. Estado del arte del problema de selección de cartera de proyectos estáticos	26
3. Descripción del problema de investigación	29
3.1. Características del problema de selección de cartera de proyectos . . .	29
3.2. Definición de la clase Problemas de selección de cartera de proyectos .	32
3.3. Retos del problema	34
4. Metodología de apoyo a la decisión	36
4.1. Hiperheurístico en que se basa el algoritmo desarrollado	36
4.2. Hiperheurístico propuesto: HH-VNS-PP	39
4.2.1. Heurísticas de bajo nivel	39
4.2.2. Estructura general	43
4.3. Metodología de apoyo a la decisión	46
4.3.1. Identificación de las características del problema	47
4.3.2. Definir las preferencias del tomador de decisiones	47
4.3.3. Ajuste de datos de entrada	48
4.3.4. Generación del conjunto de carteras	50
4.3.5. Presentación del conjunto de carteras al tomador de decisiones y selección final	50

5. Experimentación	51
5.1. Grupo A	52
5.2. Grupo B	60
5.3. Grupo C	64
 6. Conclusiones	 75
 7. Trabajo futuro	 76

CAPÍTULO 1

INTRODUCCIÓN

En este documento se presenta la investigación del proyecto de tesis titulado: Hiperheurístico multiobjetivo basado en búsqueda de vecindad variable para problemas de selección de cartera de proyectos. En el mismo se propone una metodología de solución y apoyo a la decisión para diferentes tipos de problemas de selección de cartera de proyectos.

Está estructurado en siete capítulos. En el presente, se muestran la justificación y motivación del problema de tesis, y se expone además el diseño de la investigación. En el capítulo 2 se presenta la base teórica de los métodos y herramientas empleados en el desarrollo de la metodología de solución que se propone. Posteriormente, se pasa a la descripción del problema de investigación en el capítulo 3, donde se muestran sus características y principales retos.

En el capítulo 4 se expone la metodología de apoyo a la decisión propuesta, dentro de la cual se encuentra el hiperheurístico HH-VNS-PP. En el capítulo 5 se muestra la experimentación, describiendo las instancias empleadas, características y diferencias, así como los resultados obtenidos. En el capítulo 6 se exponen las conclusiones del presente trabajo y en el 7 lo que se quisiera lograr en trabajos futuros.

1.1 JUSTIFICACIÓN Y MOTIVACIÓN

La norma ISO 21500 [60] define un proyecto como un conjunto único de procesos que consta de actividades coordinadas y controladas, con fechas de inicio y fin, que se llevan a cabo para lograr los objetivos del proyecto. El logro de los objetivos del proyecto requiere la realización de entregables que satisfagan requisitos específicos. Además, un proyecto puede estar sujeto a múltiples restricciones, tales como tiempo, costo y recursos.

En el interior de cualquier tipo de organización se gestan diversos proyectos de inversión, algunos enfocados en alcanzar los objetivos estratégicos trazados por la alta dirección y otros derivados de situaciones no pronosticadas que requieren su cumplimiento. La administración y la gestión de los proyectos resulta difícil; es necesario categorizarlos según su alcance, los recursos a emplear, impactos generados, los actores involucrados, riesgo en la inversión y otros aspectos complejos de categorizar por su carácter subjetivo [50]. Es por ello que el problema de la optimización de portafolio o cartera de proyectos es una tarea importante y compleja, pues debe asegurar una asignación adecuada de los recursos a los proyectos, que están en correspondencia con los objetivos estratégicos de la organización, garantizando un buen aprovechamiento de los mismos.

Aunque los *Problemas de Selección de cartera de Proyectos* (PSCP) pueden modelarse y resolverse mediante métodos exactos [14, 47, 56], en la práctica, muchas veces esto no resulta viable o no es eficiente por el tiempo que toma obtener la solución. Otro elemento a tener en cuenta es la subjetividad del tomador de decisiones a cargo de la evaluación de proyectos o de la cartera de proyectos, que posee capacidad limitada con múltiples objetivos para el proceso de decisión [55]. Este individuo (o grupo de individuos), durante el proceso de selección de proyectos, puede cambiar los objetivos, o no tener ideas precisas respecto a determinadas restricciones, por lo que se requiere ofrecerle soluciones eficientes que lo ayuden en la toma de decisiones

y garanticen la racionalidad del proceso.

Un método heurístico es un procedimiento para resolver un problema de optimización bien definido mediante una aproximación intuitiva, en la que la estructura del problema se utiliza de forma inteligente para obtener una buena solución [25]. Mediante heurísticas se pueden obtener soluciones aceptables en poco tiempo y, además, son un mecanismo que sirve para explorar el problema, estudiando la complejidad de instancias y la naturaleza de soluciones, pudiendo arrojar soluciones previas que sirvan como puntos de partida para métodos exactos.

Como comenta Marti [49], en los algoritmos heurísticos las técnicas e ideas aplicadas a la resolución de un problema son específicas de este, y aunque en general pueden ser trasladadas a otros problemas, han de particularizarse en cada caso. Existen métodos heurísticos que han sido clasificados en lo que se podría denominar heurístico clásicos, mientras que hay otros que se han diseñado para resolver cuestiones muy específicas, sin posibilidad de generalización o aplicación a otros problemas similares.

Ubicados en un nivel superior, guiando el diseño de los heurísticos, se encuentran los llamados metaheurísticos. Los procedimientos metaheurísticos son una clase de métodos aproximados diseñados para resolver problemas difíciles de optimización combinatoria en los que los heurísticos clásicos no son efectivos. Los metaheurísticos proporcionan un marco general para crear nuevos algoritmos híbridos, combinando diferentes conceptos derivados de la inteligencia artificial, la evolución biológica y los mecanismos estadísticos [52]. Una metaheurística es un conjunto de conceptos que se pueden utilizar para definir métodos heurísticos, pudiéndose aplicar a un amplio conjunto de diferentes problemas [10].

Existen varias metaheurísticas que han sido diseñadas para resolver problemas de selección cartera de proyectos, obteniendo resultados satisfactorios. Se pueden mencionar como ejemplos algoritmos de colonias de abejas [36], método de las restricciones, técnicas multicriterio [68], redes neuronales artificiales [18], optimización

de enjambre de partículas [51], entre otros. Estos algoritmos tienen en común que intentan encontrar las mejores soluciones a PSCP con características específicas, por lo que solo pueden emplearse para resolver el tipo de problemas para el que fueron diseñados.

En la práctica, los PSCP varían de una organización a otra, e incluso dentro de una misma organización. En algunas organizaciones se tienen que resolver PSCP de naturaleza diferente con mucha frecuencia. En estos casos no se requiere un algoritmo potente que ofrezca la mejor solución a un problema en específico, sino un algoritmo que pueda ofrecer soluciones buenas a PSCP con características diferentes, pues no es viable crear un nuevo algoritmo ante cada tipo de PSCP a resolver. Es por ello que resulta pertinente automatizar la tarea de elegir las heurísticas más adecuadas en cada caso, en vez de hacerlo manualmente.

Un hiperheurístico es, precisamente, un método de búsqueda heurística (a menudo una metaheurística) que automatiza el proceso de seleccionar, combinar, generar o adaptar varias heurísticas más simples, para resolver eficientemente problemas de búsqueda computacional, con la particularidad de ser capaz de manejar distintos tipos de problemas, de manera rápida y ofreciendo buenas soluciones, en lugar de resolver un solo tipo problema con una solución casi óptima u óptima [53].

En este trabajo se propone una aproximación metodológica a la solución integral de diferentes tipos de problemas de selección de cartera de proyectos desarrollando una metodología de apoyo a la decisión multiobjetivo empleando un algoritmo hiperheurístico.

1.2 DISEÑO METODOLÓGICO

A continuación se presenta el diseño metodológico de la investigación.

1.2.1 PROBLEMA

¿Cómo apoyar la toma de decisiones en la solución de diferentes tipos de problemas de selección de cartera de proyectos empleando un algoritmo que permita resolver estos diferentes tipos y facilitar así la implementación de una herramienta de apoyo a la decisión general?

1.2.2 OBJETIVOS E HIPÓTESIS

El objetivo general de la presente tesis es realizar una aproximación metodológica a la solución integral de diferentes tipos de problemas de cartera de proyectos desde la óptica de toma de decisiones multicriterio.

Objetivos específicos:

- Desarrollar una metodología de solución y apoyo a la decisión para diferentes tipos de problemas de selección de cartera de proyectos.
- Implementar un prototipo de solución para varios tipos de PSCP con un hiperheurístico basado en la búsqueda de vecindad variable.

Hipótesis: Si se desarrolla un hiperheurístico basado en búsqueda de vecindad variable siguiendo una metodología de solución y apoyo a la decisión para diferentes tipos de PSCP, se podrían obtener buenas soluciones para diferentes tipos de PSCP, con un único algoritmo.

1.2.3 NOVEDAD CIENTÍFICA

La novedad de este trabajo radica en la propuesta de una metodología de solución integral para diferentes tipos de problemas de selección de cartera de pro-

yectos, desde la óptica de toma de decisiones multicriterio, para lo cual se propone y desarrolla el hiperheurístico HH-VNS-PP.

1.2.4 ALCANCE Y LIMITACIONES

Aunque la naturaleza del hiperheurístico permita utilizarlo para resolver otros problemas, la metodología propuesta está enfocada a la resolución de problemas de selección de cartera de proyectos. Si se deseara resolver otros problemas habría que hacer modificaciones a la metodología y desarrollar heurísticas de bajo nivel específicas para el nuevo problema.

CAPÍTULO 2

MARCO TEÓRICO

En este capítulo se presenta la base teórica de los métodos y herramientas empleados en el desarrollo de la metodología de solución que se propone.

El capítulo está estructurado de la siguiente forma: Clasificación de los problemas de decisión (sección 2.1), Proceso de apoyo a la toma de decisión multicriterio (sección 2.2) Métodos toma de decisiones bajo múltiples criterios (sección 2.3), Optimización multiobjetivo (sección 2.4), Optimización lineal entera (sección 2.5), Hiperheurísticos (sección 2.6), Búsqueda de vecindad variable (sección 2.7) y para finalizar, se presenta el estado del arte del problema de selección de cartera de proyectos estáticos (sección 2.8).

2.1 CLASIFICACIÓN DE LOS PROBLEMAS DE DECISIÓN

Los problemas de decisión se clasifican según su problemática, según la naturaleza de las consecuencias asociadas a las alternativas, según la cardinalidad del conjunto de alternativas y según la cantidad de tomadores de decisiones [23].

- Según su problemática [61]:

Problemas de selección: El tomador de decisiones debe escoger la mejor o

las mejores alternativas, de un conjunto de alternativas, en base a sus preferencias.

Problemas de clasificación: El tomador de decisiones debe asignar cada alternativa a una única categoría previamente definida, tomando en cuenta sus preferencias.

Problemas de jerarquización: El tomador de decisiones debe ordenar, según una relación de preferencia, las alternativas a través de una jerarquización.

- Según la naturaleza de las consecuencias asociadas a las alternativas:

Decisiones bajo certeza: El tomador de decisiones conoce por adelantado las consecuencias de su decisión.

Decisiones bajo incertidumbre: El tomador de decisiones no conoce de antemano las consecuencias de elegir una alternativa, pero se conoce una distribución de probabilidad asociada a la ocurrencia de las consecuencias asociadas a la alternativa.

Decisiones bajo estricta incertidumbre: El tomador de decisiones no tiene conocimiento sobre la ocurrencia de las consecuencias asociadas a la alternativa.

- Según la cardinalidad del conjunto de alternativas:

Finito: Número pequeño.

Infinito: Número muy grande o infinito contable o infinito incontable.

- Según la cantidad de tomadores de decisiones:

Una sola persona: Las decisiones recaen sobre una sola persona

Un grupo de personas: Las decisiones son tomadas por dos o más personas.

2.2 PROCESO DE APOYO A LA TOMA DE DECISIÓN MULTICRITERIO

En la toma de decisiones multicriterio, el tomador de decisiones debe seleccionar, jerarquizar o asignar a categorías, alternativas de un conjunto denominado conjunto de elección. Para escoger en este conjunto de elección el decisor tiene diversos puntos de vista, denominados criterios. Estos criterios son, al menos, parcialmente contradictorios, en el sentido de que si el decisor adopta uno de dichos puntos de vista, no escogerá la misma alternativa que si se basa en otro criterio [6].

Teniendo esto en cuenta, de manera general, el proceso de toma de decisión puede concebirse como la elección por parte de un decisor de *lo mejor* entre *lo posible* [58].

La existencia de recursos limitados generan las restricciones del problema. Los valores de las variables de decisión que satisfacen las restricciones constituyen lo que se denomina el conjunto factible o alcanzable, que estructura y formaliza lo que se entiende por lo posible. Este conjunto puede ser continuo o discreto.

Para definir *lo mejor*, se define una función de criterio, usualmente llamada función objetivo, o varias, que pueden representarse por un sistema relacional de preferencias. Mediante el empleo de técnicas matemáticas se optimiza la función de utilidad sobre el subconjunto alcanzable, obteniendo de esta manera la mejor solución.

En la primera fase del proceso decisional requiere una información exclusivamente de tipo técnico, o sea, para determinar el conjunto factible sólo se necesita información no preferencial. Las preferencias reales del tomador de decisiones aparecen en la segunda fase.

Es importante destacar que, aunque en investigación de operaciones también

se tiene un conjunto factible y una o varias funciones objetivo, la diferencia radica en que las preferencias del tomador de decisiones no son importantes o incluso, no se toman en consideración. Por otra parte, en el apoyo a la toma de decisiones multicriterio las preferencias del tomador de decisiones son fundamentales y buena parte del proceso se centra en obtener las mismas.

El proceso de toma de decisiones no ocurre de manera continua, sino que está formado por diversas actividades que ocurren en momentos diferentes, la existencia de diferentes etapas y los diferentes requisitos en cada una de ellas, indican que los sistemas deben tener capacidades internas múltiples y deben ser flexibles y fáciles de usar. Simon [64] estableció en 1960, cuatro etapas en la toma de decisiones:

- **Inteligencia:** Se identifica la problemática, indica por qué, dónde y con qué efectos ocurre una situación.
- **Diseño:** Se diseñan las posibles alternativas de solución para un problema.
- **Selección:** Consiste en elegir una (o varias) de las alternativas propuestas.
- **Implementación:** Se lleva la decisión a la acción y se informa sobre el progreso y la solución.

En la práctica, los decisores no desean ordenar las soluciones factibles de acuerdo a un solo criterio, sino que desean efectuar esta tarea con respecto a diferentes criterios que reflejen sus particulares preferencias. Generalmente estos criterios u objetivos, entran en conflicto entre sí, por lo que se hace necesario encontrar un compromiso o equilibrio entre ellos.

De manera general los problemas de decisión multicriterio se caracterizan por tener:

Alternativas: Un número finito de alternativas, también llamadas opciones, acciones o candidatos.

Múltiples criterios: Cada problema tiene múltiples criterios, el número de criterios depende de la naturaleza del problema.

Unidades incommensurables: Cada criterio tiene diferentes medidas.

Atributos de peso: Casi todos los métodos de toma de decisiones con múltiples criterios requieren información respecto a la importancia de cada criterio, la cual es generalmente proporcionada por una escala ordinal o cardinal.

Matriz de decisión: Un problema de toma de decisión multicriterio puede ser expresado en una matriz, donde las columnas indican los criterios y los renglones las alternativas.

2.3 MÉTODOS TOMA DE DECISIONES BAJO MÚLTIPLES CRITERIOS

Existen varios métodos de toma de decisión multicriterio. A continuación se describen brevemente algunos de ellos, Velasquez y Hester [71] realizan un análisis a mayor profundidad.

- MAUT (Multi-Attribute Utility Theory) [44]: Es una teoría de utilidad esperada que puede decidir el mejor curso de acción en un problema dado asignando una utilidad a cada posible consecuencia y calcular la mejor utilidad posible.

La mayor ventaja de MAUT es que tiene en cuenta la incertidumbre. Suele tener una utilidad asignada, que no es una cualidad que se tenga en cuenta en muchos métodos MCDM. Es integral y puede explicar e incorporar las preferencias de cada consecuencia en cada paso del método.

Como desventaja, se necesita gran cantidad de información en cada paso del procedimiento, además de registrar con precisión las preferencias del tomador de decisiones, que debe dar pesos específicos a cada una de las consecuencias

en cada nivel, y esto es algo que no es posible en todos los problemas de toma de decisiones.

- AHP (Analytic Hierarchy Process) [62]: Es un método basado en la evaluación de diferentes criterios que permiten jerarquizar un proceso. Usa comparaciones por pares, que se utilizan tanto para comparar las alternativas con respecto a los diversos criterios como para estimar los pesos de los criterios.

Una de sus ventajas es su facilidad de uso. Las comparaciones por pares puede permitir que quienes toman las decisiones asignar coeficientes de peso al comparar alternativas con relativa facilidad. Es escalable y puede ajustarse fácilmente en tamaño.

Su principal desventaja es la susceptibilidad a la inversión de rango. Debido a la naturaleza de las comparaciones para las clasificaciones, la adición de alternativas al final del proceso podría hacer que las clasificaciones finales cambien o se inviertan.

- Teoría de conjuntos difusos [73]: Debido a que permite trabajar con datos imprecisos e inciertos, puede ser utilizado para resolver problemas de gran complejidad.

Su principal desventaja es que los sistemas difusos suelen ser difíciles de desarrollar. En muchos casos, pueden requerir numerosas simulaciones antes de poder usarse en el mundo real.

- CBR (Case-Based Reasoning) [22]: Este método utiliza información registrada en bases de datos y propone una solución a un problema de toma de decisión basado en los casos más similares [22].

Requiere poco esfuerzo en términos de adquisición de datos adicionales, pues se basa en datos existentes. La mayor de sus ventajas es que puede mejorar con el tiempo, a medida que se agregan más casos a la base de datos, lo que lo hace adaptable a cambios en el entorno.

Su principal inconveniente es su sensibilidad a la inconsistencia en los datos. Los casos anteriores pueden ser inválidos o los casos especiales pueden dar como resultado respuestas no válidas. Es por ello que CBR se utiliza en escenarios donde existe un número considerable de casos anteriores, como son empresas de seguros y compañías farmacéuticas.

- DEA (Data Envelopment Analysis) [19]: Utiliza técnicas de programación lineal para medir la eficiencia relativa de alternativas. Califica las eficiencias de las alternativas entre sí, con la alternativa más eficiente teniendo una calificación de 1.0, con todas las otras alternativas siendo una fracción de 1.0.

Es capaz de manejo de múltiples entradas y salidas. La eficiencia puede ser analizada y cuantificada. Permite descubrir relaciones que pudiesen pasar desapercibidas por otros métodos.

Una desventaja importante es que no trata datos imprecisos. Supone que toda la información y los datos de salida son conocidos en su totalidad y esto no ocurre siempre. Los resultados pueden ser sensibles dependiendo de las entradas y salidas

- SMART (Simple Multi-Attribute Rating Technique) [57]: Es un método simple que solo requiere independencia de utilidad e independencia preferencial. Permite cualquier tipo de técnicas de asignación de peso, es decir, relativas, absolutas, etc; y requiere menos esfuerzo por parte del tomador de decisiones. No obstante, puede no ser conveniente para determinados problemas demasiado complejos.

- SAW (Simple Additive Weighting) [43]: En el método de sumas ponderadas se asigna un valor de peso a cada uno de los objetivos y se multiplica por estos. Tiene la habilidad de compensar entre criterios. Es intuitivo para los tomadores de decisiones y el cálculo es simple, se puede realizar sin la ayuda de complejos programas computacionales.

Entre las desventajas que posee, al evaluar un objetivo no se tiene en cuenta si es de tipo maximización o minimización, se limita a convertir todo objetivo en un solo tipo. Las estimaciones reveladas no siempre reflejan la situación real, por lo que el resultado obtenido puede no ser lógico.

- Programación por Metas [59]: Es un método de programación pragmático que puede elegir entre un número infinito de alternativas, por lo que su mayor ventaja es que tiene la capacidad de manejar problemas a gran escala.

Una desventaja importante es su incapacidad para establecer coeficientes de peso. Es por esta razón que muchas aplicaciones consideran necesario combinarla con otros métodos, como el AHP, para definir los pesos.

- ELECTRE (ELimination and Choice Translating REality) [35]: Es un método de superación basado en el análisis de concordancia que emplea cuatro relaciones binarias de superación: preferencia estricta, preferencia débil, indiferencia e incomparabilidad. Partiendo de una matriz de decisión, se utiliza la relación de sobrecalificación con el fin de obtener una matriz de superación. Permite reducir el tamaño del conjunto de soluciones eficientes dividiéndolo en alternativas favorables y alternativas no favorables para el decisor, aunque también permite ordenarlas.

Al basarse en el principio del voto mayoritario y la teoría de la votación, su principal ventaja es que puede ser explicado de forma sencilla a los tomadores de decisión, quienes en general comprenden sin mucho esfuerzo los principios de los métodos.

Su mayor desventaja es la asignación de la importancia relativa a los criterios o atributos que se expresa en una escala cardinal (refleja la intensidad de las preferencias). O sea, si un atributo A tiene un peso asignado de 0.2 y un atributo B tiene un peso asignado de 0.1, eso significa que A es 2 veces más importante que B para el tomador de decisiones.

- PROMETHEE [11]: Es similar a ELECTRE en que también tiene varias ite-

raciones y es un método de superación.

Su ventaja es que es fácil de usar. No requiere suponer que los criterios son proporcionados.

Las desventajas son que no proporciona un método claro para asignar pesos y requiere la asignación de valores, pero tampoco proporciona un método claro para asignarlos.

Su facilidad de uso lo ha convertido en un método común ya que sus iteraciones han ido mejorando.

- TOPSIS (Technique for Order of Preference by Similarity to Ideal Solution) [70]: Es un enfoque para identificar la alternativa más cercana a la solución ideal y más lejana a la negativa solución ideal en un espacio multidimensional. Tiene numerosas ventajas.

Es fácil de usar y programable. El número de pasos permanece igual independientemente del número de atributos.

Una desventaja es que su uso de la distancia euclidiana no considera la correlación de los atributos. Es difícil ponderar los atributos y mantener la coherencia de juicio a medida que aumenta el número de atributos.

2.4 OPTIMIZACIÓN MULTIOBJETIVO

El problema de optimización multiobjetivo (MOP, por sus siglas en inglés), también denominado de *optimización vectorial* o *multicriterio*, se puede definir como la determinación de los valores para un vector de variables de decisión que satisfaga un conjunto de restricciones y optimice una función vectorial, cuyos elementos representan las funciones objetivo individuales [65]. Estas funciones forman una descripción matemática de criterios de desempeño que generalmente están en conflicto entre sí. Por tanto, el término *optimización* significa encontrar una solución que proporcione valores en todas las funciones objetivo aceptables para el decisor.

A continuación, se definen los elementos principales de un modelo de optimización multiobjetivo [5]:

- **Variables de decisión:** Conjunto de n variables cuyos valores forman una solución (puede o no ser factible) a un problema de optimización, agrupadas en un vector X y cada variable representada por x_j ; donde $j = 1, 2, \dots, n$.
- **Restricciones:** Delimitan el problema y validan las soluciones. Por lo tanto, se puede decir que las restricciones dibujan el contorno de la región donde se encuentra el conjunto factible del problema.
- **Función objetivo:** Las funciones objetivo forman el criterio de evaluación para saber que tan buena es una solución. Al igual que las restricciones, son funciones de las variables de decisión. En la optimización multiobjetivo existen dos o más funciones objetivos $(f_1(\bar{X}), f_2(\bar{X}), \dots, f_k(\bar{X}))$ en cada problema. El vector de funciones objetivo puede representarse como $f(\bar{X}) = [f_1(\bar{X}), f_2(\bar{X}), \dots, f_k(\bar{X})]^T$.

Un problema de optimización multiobjetivo es un conjunto de n variables de decisión, k funciones objetivo, m restricciones de desigualdad y p de igualdad. En estos problemas el objetivo de la optimización es encontrar el vector de decisión \bar{X} para minimizar o maximizar:

$$f(\bar{X}) = [f_1(\bar{X}), f_2(\bar{X}), \dots, f_k(\bar{X})]^T$$

cumpliendo con:

$$\bar{g}(\bar{x}) \leq 0; \bar{g} = g_1, g_2, \dots, g_m.$$

$$\bar{h}(\bar{x}) = 0; \bar{h} = h_1, h_2, \dots, h_p.$$

Generalmente, no existe la solución óptima en todos los objetivos simultáneamente. Se busca entonces un compromiso aceptable entre ellos, para lo que es necesario tener claros algunos conceptos definidos a continuación:

- **Conjunto factible:** El conjunto factible S está definido como el conjunto de vectores de decisión X que satisfaga las restricciones de desigualdad $\overline{g}_i(\overline{x})$ y las restricciones de igualdad $\overline{h}_i(\overline{x})$.
- **Dominancia de Pareto:** Para dos vectores de decisión $\overline{x}^*, \overline{y}^* \in X$ se dice que x^* domina a y^* , si y solo si
 - la solución \overline{x}^* no es peor que \overline{y}^* en ninguno de sus objetivos. En caso de minimización, esta condición se puede traducir en:

$$f_j(\overline{x}^*) \leq f_j(\overline{y}^*), \forall j = 1, 2, \dots, m;$$
 - la solución \overline{x}^* es estrictamente mejor que la solución \overline{y}^* en, al menos, un objetivo:

$$f_j(\overline{x}^*) < f_j(\overline{y}^*) \text{ para al menos un } j \in 1, 2, \dots, m;$$

lo denotaremos como $f(\overline{x}^*) \leq f(\overline{y}^*)$.

- **Óptimo de Pareto:** Una solución factible $\overline{x} \in X$ es llamada eficiente u óptima de Pareto, si no existe otra $x \in X$ tal que $f(x) \leq f(\overline{x})$. El concepto de óptimo de Pareto tiene gran importancia en optimización multiobjetivo.
- **Frontera de Pareto:** Para un MOP dado y un conjunto óptimo de Pareto P^* , la frontera de Pareto se define como:

$$\mathcal{F}^* = \{(f_1(x), f_2(x), \dots, f_k(x)) | x \in P^*\}.$$

Es decir, la frontera de Pareto está compuesta por los vectores en el espacio de objetivos que son óptimos de Pareto.
- **Solución Pareto óptima local:** Una solución x es Pareto óptima local si y solo si:

$$\forall w \in N(x), F(w) \text{ no domina a } F(x)$$

donde $N(x)$ representa la vecindad de la solución x . Algunas soluciones Pareto óptimas pueden ser obtenidas a partir de la resolución de la siguiente expresión matemática:

$$\mathcal{S}_\lambda = \min F(x) = \sum_{i=1}^n \lambda_i f_i(x),$$

$$x \in S \text{ con } \lambda_i \geq 0 \text{ para } i = 1, \dots, n \text{ y } \sum_{i=1}^n \lambda_i = 1.$$

Estas soluciones son conocidas como soluciones soportadas. Se generan a partir de la resolución de \mathcal{S}_λ utilizando diferentes vectores de pesos λ . La complejidad de \mathcal{S}_λ es la misma que la del problema de optimización mono-objetivo subyacente. Si los problemas de optimización mono-objetivo subyacentes presentan una complejidad polinomial, será relativamente sencillo generar soluciones soportadas. Sin embargo, existen otras soluciones Pareto óptimas que no pueden obtenerse resolviendo una \mathcal{S}_λ . En realidad, estas soluciones, conocidas como soluciones no soportadas, son dominadas por combinaciones convexas de soluciones apoyadas, es decir, puntos de la forma convexa $Y = F(S)$.

2.5 OPTIMIZACIÓN LINEAL ENTERA

La programación lineal es la maximización o minimización de una función lineal, denominada función objetivo. Las variables de esta función (variables de decisión) están sujetas a una serie de restricciones expresadas mediante un sistema de ecuaciones o inecuaciones también lineales.

Existen problemas lineales [8] donde es aceptable que las variables de decisión tomen valores fraccionales, por ejemplo, se pueden producir 4.1 toneladas de trigo o recorrer 3.5 km. Sin embargo, existen muchos problemas donde las soluciones fraccionales no son aceptables, como es el caso de problemas de cortes de rollos, asignación de tareas, selección de invitados a una boda, el PSCP (investigado en este trabajo), entre otros.

Un problema de programación lineal entera [63] es un problema de programación lineal con la restricción adicional de que algunas de las variables solo pueden tomar valores enteros. Una gran variedad de problemas combinatorios pueden

ser planteados como problemas de programación lineal entera.

2.5.1 CLASIFICACIÓN DE PROBLEMAS LINEALES ENTEROS

Los problemas lineales enteros pueden categorizarse atendiendo a diferentes características.

- **Atendiendo al tipo de variables:**

- **Enteros puros:** son aquellos en que todas las variables únicamente pueden tomar valores enteros. También se distinguen dentro de estos los problemas totalmente enteros como aquellos en que tanto las variables como todos los coeficientes que intervienen en el problema han de ser enteros.
- **Mixtos:** son aquellos en los que hay al mismo tiempo variables continuas y variables que sólo pueden tomar valores enteros.
- **Binarios:** las variables sólo pueden tomar los valores cero o uno.

- **Atendiendo al criterio del tipo de problema:**

- **Directo:** Si el problema de decisión involucra variables enteras.
- **Codificado:** Cuando se trata de un problema que contiene además de aspectos cuantitativos, alguna consideración de tipo cualitativos, y por ello para tratar este tipo de aspectos se requiere el uso de variable enteras o binarias.
- **Transformado:** Cuando el problema no incluye variables enteras, pero para ser tratado analíticamente requiere el uso de variable enteras artificiales.

Wolsey [72] explica el siguiente modelo matemático para programación lineal entera.

Suponga que tiene un problema representado por un modelo lineal

$$\text{máx}\{cx : Ax \leq b, x \geq 0\},$$

donde A es una matriz de $m \times n$, c es un vector renglón n -dimensional y b es un vector columna m -dimensional. A este se agregan restricciones lineales que obligan que ciertas variables sólo tomen valores enteros.

$$\text{máx } cx + hy,$$

$$Ax + Gy \leq b,$$

$$x \geq 0, y \geq 0 \text{ entera.}$$

.

2.6 HIPERHEURÍSTICOS

Un hiperheurístico es un algoritmo de alto nivel que, dada una instancia de un problema en particular y un conjunto de heurísticas de bajo nivel, es capaz de seleccionar y aplicar la heurística de bajo nivel apropiada en cada momento de decisión [13, 53]. Este término fue usado por primera vez en 1997 por Jörg Denzinger, Matthias Fuchs y Marc Fuchs [24] para describir un protocolo que selecciona y combina varios métodos de Inteligencia Artificial.

Tiempo después en el año 2000, Cowling y Soubeiga lo usaron de hiperheurístico para describir la idea de heurísticas que seleccionan heurísticas, en el contexto de optimización combinatoria [20].

Chakhlevitch [16] lo define de la siguiente forma:

1. Un hiperheurístico es una heurística de alto nivel que administra un conjunto de heurísticas de bajo nivel, con cardinalidad mayor que uno.
2. Busca un buen método para resolver el problema en lugar de una buena solución.

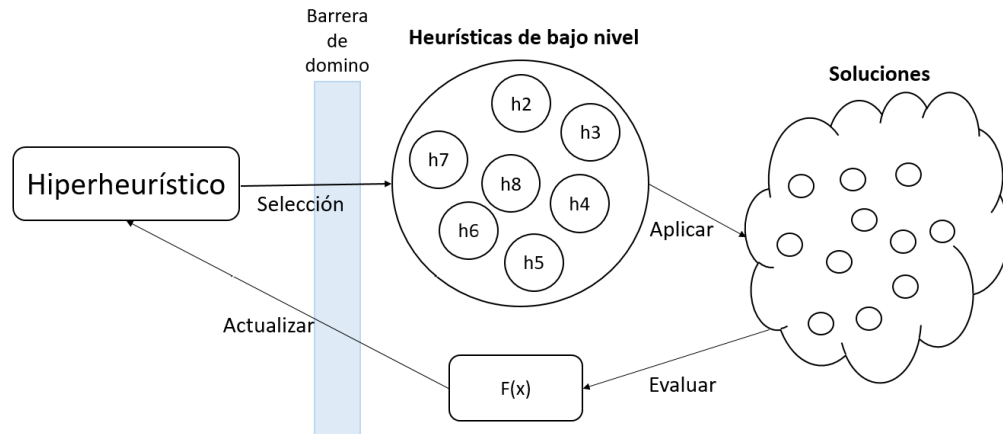


Figura 2.1: Estructura general de un hiperheurístico.

3. Utiliza solo información limitada sobre problemas específicos (idealmente, esta información incluye solo el número de heurísticas de bajo nivel para que el problema y las funciones objetivo se maximicen o minimicen).

En la figura 2.1 puede verse la estructura general de un hiperheurístico.

De acuerdo con Burke [12], las hiper-heurísticas pueden clasificarse según su fuente de retroalimentación y según la naturaleza del espacio de búsqueda heurística. Con respecto a la naturaleza del espacio de búsqueda heurística se clasifican en:

- Metodologías de selección de heurísticas: Producen combinaciones heurísticas de selección o de perturbación preexistentes.
- Metodologías de generación de heurísticas: Genera nuevos métodos heurísticos utilizando componentes básicos/bloques de construcción o búsqueda de rastro de heurísticas de construcción o perturbación existentes.

Con respecto a la fuente de retroalimentación durante el aprendizaje se clasifican en:

- Hiper-heurísticas de aprendizaje en línea: Aprende mientras resuelve una instancia determinada de un problema.

- Hiper-heurísticas de aprendizaje fuera de línea: Aprende de un conjunto de instancias de entrenamiento, un método que generalizaría a instancias no vistas.
- Hiper-heurísticas sin aprendizaje: No utiliza información previa del proceso de búsqueda del rendimiento del heurístico de bajo nivel.

2.7 BÚSQUEDA DE VECINDAD VARIABLE

La búsqueda local es la base de muchos de los métodos usados en problemas de optimización [1, 17, 48, 54]. Es un proceso iterativo que partiendo de una solución inicial, busca en su vecindad una mejor solución. Si la encuentra, reemplaza su solución actual por la nueva y continúa con el proceso, hasta que no se pueda mejorar la solución actual. Esto puede observarse en el algoritmo 1, que al igual que el resto de los ejemplos del epígrafe, toma el caso de minimización.

Algoritmo 1 Búsqueda local.

```

 $x \leftarrow$  Solución inicial
MáximoLocal  $\leftarrow$  falso
mientras MáximoLocal = falso hacer
  MáximoLocal  $\leftarrow$  cierto
  para  $x'$  en la vecindad de  $x$  hacer
    si  $f(x') < f(x)$  entonces
       $x \leftarrow x'$ 
    fin si
  fin para
fin mientras
devolver  $x$ 

```

La vecindad de una solución x es el conjunto de todas aquellas soluciones x' que pueden ser alcanzables a partir de x por medio de un movimiento. Un movimiento

puede ser una inserción, eliminación o intercambio de componentes en una solución, entre otros.

El cómo se busca la vecindad y cuál vecino se usa en el reemplazo, en general puede ser:

- Seleccionar el mejor vecino de todos.
- Seleccionar el primer vecino que mejora la solución.

La búsqueda local tiene la ventaja de encontrar soluciones muy rápidamente. Sus principales desventajas son que queda atrapada fácilmente en óptimos locales que no son globales en la mayoría de los casos y que su solución final depende fuertemente de la solución inicial. Es por ello que no suele usarse de forma aislada, sino que se integra en otras estrategias, entre las que se encuentra la *búsqueda local con vecindad variable* (VNS).

La VNS fue presentada por Hansen y Mladenovi [37] en 1997. De manera general, parte de tener un conjunto finito de vecindades sobre las cuales se va realizando la búsqueda local.

Sea N_k un conjunto finito de vecindades predefinidas y $N_k(x)$ el conjunto de soluciones en la k -ésima vecindad de x . La vecindad cambia de forma sistemática, teniendo en cuenta tres hechos simples [38]:

1. Un mínimo local con una estructura de vecindad, no lo es necesariamente con otra.
2. Un mínimo global es el menor mínimo local que se puede encontrar en todas las posibles vecindades.
3. Para muchos problemas, los mínimos locales están relativamente próximos entre sí.

Los dos primeros hechos sugieren el empleo de varias estructuras de vecindad en las búsquedas locales para abordar un problema. El último hecho, constatado empíricamente, indica que los óptimos locales proporcionan información acerca del óptimo global. Puede ser, por ejemplo, que tengan características comunes pero, generalmente, no se sabe cuales son esas características. Es conveniente, por tanto, realizar un análisis de las proximidades de cualquier óptimo local buscando información que permita orientar la búsqueda hacia el óptimo global.

En el algoritmo 2, se puede ver un ejemplo de un a función de cambio de vecindad. La función compara el incumbente $f(x)$ con el nuevo valor $f(x')$ obtenido del k -ésimo vecindario. Si se obtiene una mejora, se actualiza el incumbente y k retorna a su valor inicial; si no, se pasa a otro vecindario.

Algoritmo 2 Cambio de vecindad.

Función: CambioVecindad (x, x', k)

si $f(x') < f(x)$ **entonces**

$x \leftarrow x'$

$k \leftarrow 1$

si no $k \leftarrow k + 1$

fin si

devolver x, k

Al momento de desarrollar el algoritmo, se pueden tomar tres enfoques: determinístico, estocástico, o una combinación de las dos. Teniendo esto en cuenta, se han desarrollado varias variantes de VNS.

Un ejemplo de enfoque determinístico es el VND (*Variable neighborhood descent*, por siglas en inglés) [39]. En este caso, la idea es buscar sistemáticamente en diferentes esquemas de vecindad, empleando un orden secuencial, hasta llegar a un mínimo local que es mínimo con respecto a todas las vecindades (ver algoritmo 3).

Con enfoque estocástico, se puede mencionar el RVNS (*Reduced VNS*). En este caso, se seleccionan puntos aleatoriamente de $N_k(x)$. Los valores de estos puntos

Algoritmo 3 VND.

Función: VND(x, k_{max}) $k \leftarrow 1$ **repetir** $x' \leftarrow$ Mejor vecino en $N_k(x)$ $x, k \leftarrow$ CambioVecindad (x, x', k)**hasta que** $k = k_{max}$ **devolver** x

son comparados con el incumbente, y este se actualiza en caso de mejora. Este algoritmo suele utilizarse en instancias muy grandes, en las que la búsqueda local es muy costosa, por lo que suele tener además, alguna condición de parada, como el máximo número de iteraciones entre mejoras o el máximo tiempo transcurrido. En el algoritmo 4 puede verse un ejemplo que emplea el tiempo de CPU como condición de parada. La función Agitar genera aleatoriamente un punto del k -ésimo vecindario.

Algoritmo 4 RVNS.

Función: RVNS(x, k_{max}, t_{max})**repetir** $k \leftarrow 1$ **repetir** $x' \leftarrow$ Agitar(x, k) $x, k \leftarrow$ CambioVecindad(x, x', k)**hasta que** $k = k_{max}$ $t \leftarrow$ TiempoCPU()**hasta que** $t > t_{max}$ **devolver** x

2.8 ESTADO DEL ARTE DEL PROBLEMA DE SELECCIÓN DE CARTERA DE PROYECTOS ESTÁTICOS

La selección de cartera de proyectos y la asignación de recursos es una tarea que se lleva a cabo en organizaciones del sector público y privado. En la mayoría de los casos, al seleccionar la cartera, el sector privado busca maximizar las ganancias mientras que en el sector público se consideran criterios asociados al beneficio o impacto social. Los PSCP se han intentado resolver de diversas maneras. Inicialmente, los decisores lo hacían de manera intuitiva, escogiendo aquellos proyectos que creían más adecuados, según sus prioridades y presupuesto. Poco a poco, empezaron a incluir metodologías o sencillos modelos matemáticos que ayudaban a evaluar cada uno de los proyectos, y luego los seleccionaban en orden hasta agotar el presupuesto.

A partir de finales de los 90 comienzan a aparecer estudios que establecen que lo más eficiente para una organización no es utilizar un modelo que seleccione los mejores proyectos, sino el que permita seleccionar la cartera de proyectos que aproveche las sinergias existentes al realizar más de un proyecto a la vez, y resuelvan las necesidades de la organización.

En el año 2000, Ghasemzadeh y Archer [34] consideran para la resolución de PSCP una única función objetivo ponderada. Sin embargo, sus instancias de prueba fueron muy limitadas porque aspiraban a una comparación entre las carteras calculadas manualmente y las construidas empleando su sistema de soporte de decisiones.

Para resolver problemas que emplean restricciones adicionales, posteriormente se propone un procedimiento de dos etapas [67]. Durante la primera fase, se construye la frontera de Pareto de carteras de proyectos eficientes, a través de la optimización. Luego, en la segunda fase, los tomadores de decisiones lo exploran de forma interactiva para tener en cuenta las preferencias personales. Además, tienen en cuenta las limitaciones de piso y techo para la inclusión de proyectos de cualquier subconjunto

dado, así como las limitaciones de recursos y los requisitos mínimos de beneficios para proyectos individuales. Como existen posibles sinergias entre los proyectos que deben evaluarse para estimar con precisión los beneficios de una cartera de proyectos, los autores intentan incorporar estas consideraciones en su metodología.

En 2014 Doerner et al. [26] mejoran más el enfoque de optimización de colonias de hormigas de Pareto. Su modelo mejorado funciona mejor con muchas funciones objetivo y un amplio conjunto de soluciones eficientes y, por lo tanto, es específicamente adecuado para problemas de la vida real.

Con el paso de los años la tendencia ha sido el resolver problemas cada vez más complejos, incluyendo aspectos difíciles de modelar como el riesgo, con la mayor precisión posible [3, 30, 46]. Debido a que los algoritmos de solución están enfocados en resolver un tipo de problema en específico, que además suele ser bastante complejo, de la mejor manera, esto hace que solo puedan emplearse para problemas con características muy similares.

En el cuadro 2.1 se muestran algunas de las investigaciones encontradas en la búsqueda bibliográfica y las características de los problemas que resuelven. La asignación total de recursos, es mayormente considerada en los trabajos de selección de cartera, mientras que el financiamiento parcial de recursos se aborda escasamente, al igual que la modelación a nivel de tareas de proyectos.

Cuadro 2.1: Literatura del problema de selección de cartera de proyectos.

Fuente	Multiobjetivo	Tareas	Asignación Total	Asignación Parcial	Sinergias
[28]	Sí	No	Sí	No	Sí
[29]	Sí	No	Sí	Sí	No
[47]	Sí	Sí	Sí	No	No
[66]	No	No	Sí	Sí	No
[9]	Sí	No	Sí	No	Sí
[45]	Sí	No	Sí	No	No
[33]	No	No	Sí	No	Sí
[7]	Sí	No	Sí	No	No
[30]	Sí	No	Sí	No	Sí
[40]	No	No	Sí	No	Sí
[21]	Sí	No	Sí	No	No
[74]	Sí	No	Sí	Sí	Sí
[4]	Sí	Sí	Sí	Sí	Sí
[31]	Sí	No	Sí	Sí	Sí
[2]	Sí	Sí	Sí	Sí	Sí
[50]	No	Sí	Sí	No	Sí

CAPÍTULO 3

DESCRIPCIÓN DEL PROBLEMA DE INVESTIGACIÓN

En este capítulo se describe con mayor profundidad el problema de investigación del presente trabajo, mostrando sus características y principales retos.

3.1 CARACTERÍSTICAS DEL PROBLEMA DE SELECCIÓN DE CARTERA DE PROYECTOS

En 1999 Dye y Pennypacker [27] definen una cartera de proyectos como aquel grupo de proyectos que, llevados a cabo bajo la dirección y patrocinio de la organización, comparten unos recursos escasos en un periodo de tiempo determinado para satisfacer un bien común, en este caso, los objetivos de la organización. Los proyectos en este trabajo son de investigación y desarrollo o de impacto social, que tienen la característica de comenzar y terminar en el mismo periodo.

De manera similar, Junguito, Perfetti y Becerra [42] la definen en 2014 como una colección de proyectos y/o programas que se agrupan juntos para facilitar la gestión efectiva y así alcanzar los objetivos estratégicos de la organización. Plantea que los componentes de la cartera son cuantificables y pueden medirse,

clasificarse y priorizarse.

Al seleccionar una cartera de proyectos eficiente es necesario tener en cuenta varios factores o aspectos fundamentales [15]:

- La consideración de diversos objetivos de manera simultánea (problema multiobjetivo).
- La interacción o dependencia entre proyectos, permitiendo valorar todo tipo de relaciones entre proyectos (sinergias positivas o negativas, relaciones de precedencia, complementariedad entre proyectos, etc.).
- El tiempo como aspecto fundamental para analizar el problema desde una vertiente más completa.
- El análisis del riesgo medido como fracaso o falta de éxito de los proyectos seleccionados.
- Pueden considerarse actividades asociadas a los proyectos, que se utilizan para la planeación y la asignación de recursos.

Teniendo esto en cuenta, podemos clasificar los problemas de selección de cartera de proyectos, de acuerdo a sus características, según los siguientes aspectos [28]:

- Vigencia en la cartera:
 - Estáticos: Todas las propuestas que son financiadas inician y terminan en un mismo periodo.
 - Dinámicos: Es posible incluir o abandonar un proyecto en diferentes periodos.

- **Asignación:**
 - **Total:** Cada proyecto solicita un presupuesto y este es asignado completamente o no.
 - **Parcial:** Cada proyecto solicita el presupuesto dentro de un rango. Si se asigna el máximo solicitado, el beneficio esperado es el máximo. Si se asigna un presupuesto entre el mínimo y el máximo, el beneficio esperado disminuye en una proporción dada. No se puede asignar un presupuesto menor al mínimo solicitado o el proyecto no generará beneficios.
- **Organización:**
 - **Regiones:** Los proyectos están organizados por regiones geográficas. Generalmente se asigna determinado rango de presupuesto para cada región, de acuerdo a las preferencias del tomador de decisiones.
 - **Áreas:** Los proyectos se agrupan según el área que benefician. Por ejemplo, en proyectos educativos, puede referirse a: humanidades, ingenierías, ciencias exactas, etc. Dentro de una empresa que manufacture autos, las áreas pudiesen ser: carrocería, interior, motor, suspensión, etc. Sea cual fuere el caso, a cada área se le asigna un rango de presupuesto, de modo que los proyectos seleccionados entren en ese rango.
- **Granularidad[2]:**
 - **Proyectos:** Esta es la más sencilla de las granularidades, los proyectos son considerados como tal, sin subdivisiones o interdependencias entre ellos.
 - **Tareas:** Los proyectos se subdividen en tareas, que pueden tener mayor o menor impacto dentro del proyecto, así como necesitar cantidades diferentes de recursos.
 - **Sinergias:** Consisten en efectos ya sea de recursos o de beneficios, que se producen por relaciones de proyectos y/o tareas de proyectos. Por ejemplo,

el ahorro de recursos por la compra de algún equipo de cómputo que pueda ser compartido .

3.2 DEFINICIÓN DE LA CLASE PROBLEMAS DE SELECCIÓN DE CARTERA DE PROYECTOS

En esta sección se define una estructura algebraica para la clase de problemas de selección de carteras de proyectos que se abordará en esta investigación. En base a las características descritas en la sección anterior, se define la clase de problemas de selección de carteras de proyectos atendiendo a las características mostradas en el cuadro 3.1.

Cuadro 3.1: Estructura de la clase Problema de selección de cartera de proyectos.

Vigencia en la cartera	Dinámicos	No
	Estáticos	Sí
Asignación	Total	Una u otra
	Parcial	
Organización	Regiones	Cualquier combinación incluyendo ninguna
	Áreas	
	Otro tipo	
Estructura de proyectos	Única	Una u otra
	Divididos en tareas o actividades	
Sinergias	Entre proyectos	Cualquier combinación incluyendo ninguna
	Entre tareas o actividades	
Información de preferencia	Con información	Una u otra
	Sin información	

La Red de Decisión y Optimización Multicriterio (OPTISAD) ha realizado varias investigaciones relacionadas con el problema de selección de cartera de proyectos. A continuación se comenta sobre algunos de los resultados obtenidos hasta el momento, haciendo énfasis los tipos de problemas que intentan resolver.

En 2010 Litvinchev y otros [47] se propone un modelo para problemas de selección de cartera de proyectos públicos a gran escala, obteniendo soluciones a problemas de hasta 25000 proyectos. Las características fundamentales de el problema tratado son las siguientes:

- Se conoce el presupuesto disponible para financiar los proyectos y se anuncia una convocatoria para solicitar propuestas en diversas áreas (por ejemplo, áreas de conocimiento), cada área con un límite inferior y superior de presupuesto.
- La decisión sobre la financiación se toma una vez por período, o sea, en cuanto a vigencia en la cartera, es estático.
- Las propuestas se originan en instituciones mutuamente no relacionados, por lo tanto, los proyectos se consideran estadísticamente independientes con correlación muy pequeña o cero (no existen sinergias entre los proyectos).
- Se desea que la cartera elegida mantenga un compromiso en cuanto a beneficio y número de proyectos (problema biobjetivo).

En 2013 Bastiani y otros [7] proponen un modelo de selección de cartera de proyectos formulado por un conjunto de indicadores que reflejan el impacto de la cartera en términos de número de proyectos y los puestos que ocupan en una clasificación realizada previamente por el tomador de decisiones. Se emplea un algoritmo de optimización evolutiva multiobjetivo empleando una medida de impacto vectorial. Dan solución a un problema estático, con asignación total.

Cruz y otros en 2014 [21] ofrecen una propuesta de solución a problemas de selección de cartera de proyectos multiobjetivo, considerando las preferencias del

tomador de decisiones a priori. Este trabajo introduce sinergias entre proyectos y consideran asignación total. Se resuelven instancias de hasta 9 objetivos.

Arratia y colegas [3] en 2018, proponen un modelo para resolver problemas biobjetivos, donde los proyectos son granulados en tareas con política de asignación total y pueden existir sinergias entre ellas.

Las investigaciones mencionadas trabajan problemas de proyectos estáticos. Otra característica que tienen en común es que cada uno intenta ofrecer la mejor solución posible al tipo de problema que resuelve, pero, a menos que se realicen modificaciones a los algoritmos o modelos que proponen, ninguno sería capaz de resolver el tipo de problema que resuelve otro.

Precisamente, este trabajo propone el algoritmo HH-VNS-PP, que, si bien no pretende encontrar la solución óptima o cercana a la óptima, ofrece buenas soluciones a cualquiera de los problemas antes mencionados, o a cualquier otro que tenga una combinación diferente de esas características.

3.3 RETOS DEL PROBLEMA

El resolver diferentes tipos de problemas de selección de cartera de proyectos con un único algoritmo presenta varios retos. A continuación se mencionan y describen brevemente los más significativos.

- **Flexibilidad del algoritmo:** El algoritmo debe ser lo suficientemente flexible como para adaptarse a las características de cada problema. Por ejemplo, Las estrategias o acciones a desarrollar para resolver un problema bi-objetivo en el que se desea maximizar el número de proyectos en la cartera y el beneficio arrojado por estos, no son las mismas que si se desea resolver un problema en el que no importa el número de proyectos en la cartera, pero sí alcanzar un balance entre cuatro grupos diferentes a beneficiar.

- **Dificultad variable debido a la naturaleza de los problemas:** Los problemas a resolver por el algoritmo presentan diferentes dificultades, que no están en relación con el número de proyectos presentados, lo cual debe ser considerado al momento de diseñar el algoritmo.

Para ilustrar esto, consideremos dos problemas, ambos con una propuesta de 100 proyectos y presupuesto para financiar aproximadamente el 60 % de ellos. Si el primer problema es de asignación total, bi-objetivo y organizado por áreas y el segundo es de asignación parcial, organizados por áreas y regiones y tiene en cuenta cuatro objetivos, el segundo problema es más complejo de resolver, por lo tanto, el número de operaciones para resolver el primer problema, debe ser considerablemente menor que el número de operaciones para resolver el segundo, cuando ambos problemas presentan un número de variables lo suficientemente grande.

- **Bajos recursos de hardware:** Se desea que el algoritmo desarrollado no demande de grandes requerimientos de hardware, pues esto implica inversión en equipos avanzados de cómputo. El algoritmo propuesto debe poder ejecutarse en un computador personal o portátil, con prestaciones medias, por lo que debe ser lo más simple posible.
- **Tiempo de ejecución razonable:** El algoritmo debe ser capaz de entregar una solución aceptable en un tiempo razonable, de acuerdo a la dificultad del problema en cuestión y número de variables.
- **El hiperheurístico adecuado:** En la literatura se encontró una vasta referencia a hiperheurísticos, no obstante, no se encontraron referencias que aborden varios tipos de problema de selección de cartera de proyectos con un único algoritmo.

CAPÍTULO 4

METODOLOGÍA DE APOYO A LA DECISIÓN

En este capítulo se describe la metodología de apoyo a la decisión propuesta en esta investigación para la solución de diferentes problemas de selección de carteras de proyectos, con las características mencionadas al final de la sección anterior. En la metodología se incluye la descripción del hiperheurístico desarrollado (HH-VNS-PP) como generador de carteras localmente eficientes.

4.1 HIPERHEURÍSTICO EN QUE SE BASA EL ALGORITMO DESARROLLADO

El hiperheurístico propuesto se basa en el enfoque mixto de Ping-Che Hsiao, Tsung-Che Chiang y Li-Chen Fu [41] (PTL), cuyo diagrama se puede ver en la figura 4.1. Este consta de cinco etapas: agitar, búsqueda local, tabu, selección ambiental y ajuste periódico. El algoritmo posee dos listas de heurísticas de bajo nivel, H_l para almacenar las que se emplean para realizar la búsqueda local y H_a para las usadas en la etapa de agitar.

Al inicio de la búsqueda, se genera una población aleatoria de soluciones S_b

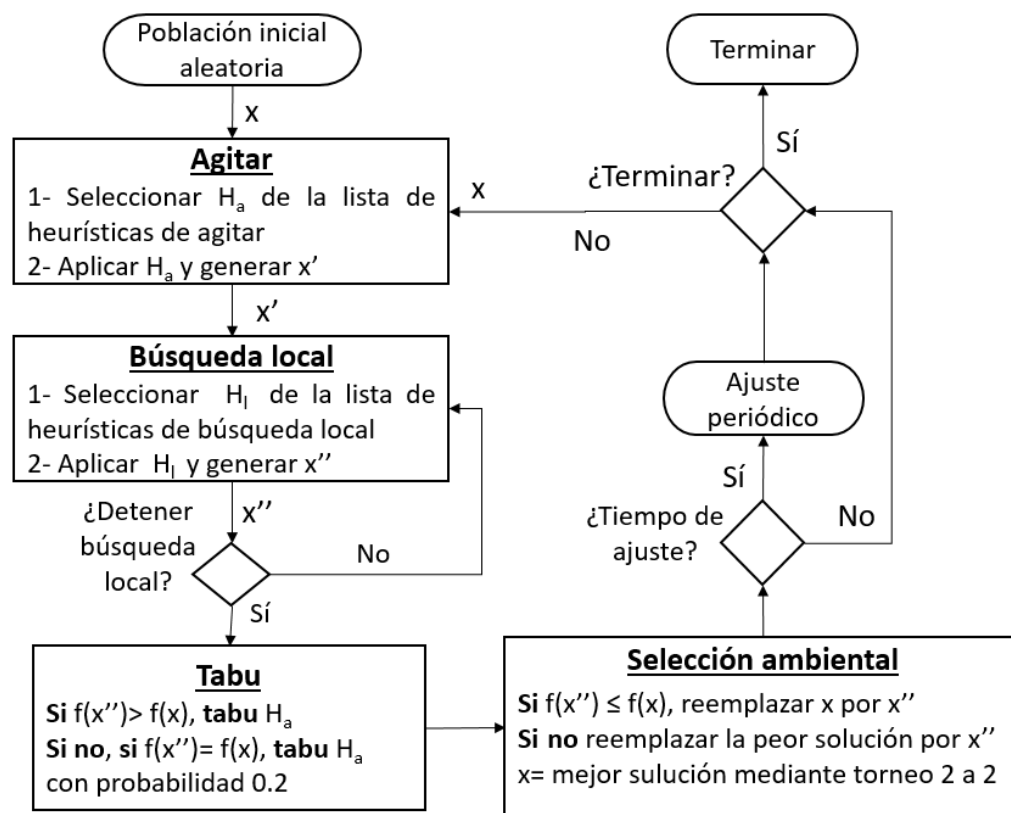


Figura 4.1: Hiperheurístico propuesto por Ping-Che Hsiao, Tsung-Che Chiang y Li-Chen Fu [41].

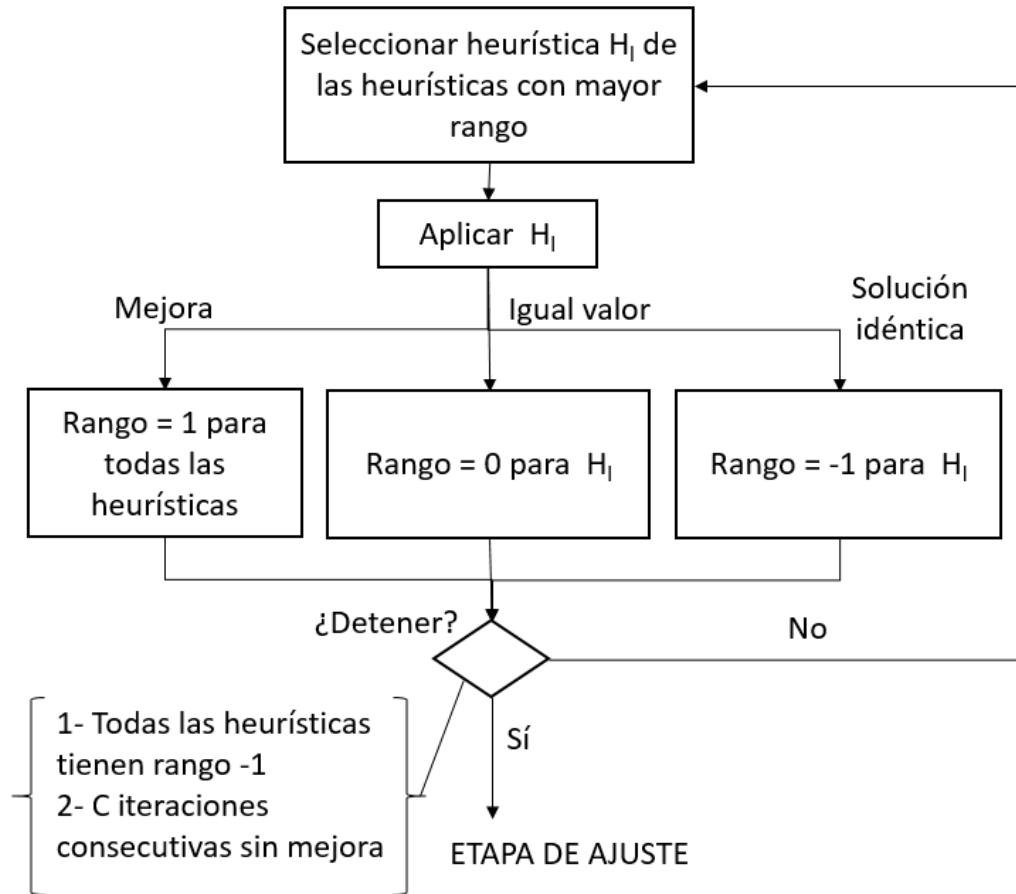


Figura 4.2: Algoritmo de búsqueda local propuesto por Ping-Che Hsiao, Tsung-Che Chiang y Li-Chen Fu [41].

por razones de diversificación. Luego de un tiempo se selecciona la mejor de esas soluciones y se usa como referencia. En la etapa de agitar, se selecciona una heurística de H_a y se aplica a x (que inicialmente es una solución aleatoria de S_b), obteniéndose x' , que será el incumbente en la etapa de búsqueda local.

Posteriormente, en la etapa de búsqueda local, se aplican las heurísticas de H_l (según las reglas de la búsqueda local establecidas en la figura 4.2) con la finalidad de encontrar un óptimo local a partir de la solución actual. Las heurísticas en esta categoría comprueban el vecindario por una mejor solución y garantizan que la solución no empeore.

La selección ambiental determina cuál de las soluciones en S_b será reemplazada y cuál será la solución base en la próxima iteración.

El ajuste periódico se activa cada vez que el algoritmo alcanza un período dado T . Los parámetros que ajustan son c (número máximo de iteraciones sin mejora permitidos en la búsqueda local) y el tamaño de la población.

PTL está diseñado para resolver problemas de un solo objetivo, mientras que el presente trabajo se propone resolver problemas de selección de carteras de proyecto, que, en su mayoría, son multiobjetivo. El algoritmo propuesto en este trabajo, HH-VNS-PP, es de naturaleza multiobjetivo y su funcionamiento es incluso más simple para poder ejecutarlo en equipos con bajos recursos de hardware, uno de los principales retos.

4.2 HIPERHEURÍSTICO PROPUESTO: HH-VNS-PP

HH-VNS-PP mantiene el principio de agitación y búsqueda local de VNS, de manera similar al de PTL. Mediante la agitación, se perturba la solución, mejorando la exploración del espacio de búsqueda y con ello la fiabilidad global. Mientras que, en el paso de búsqueda local se mejora la profundidad de búsqueda de una solución, con el objetivo de alcanzar un óptimo local. Esto puede observarse en la figura 4.3.

4.2.1 HEURÍSTICAS DE BAJO NIVEL

HH-VNS-PP agrupa las heurísticas de bajo nivel en dos categorías: agitación y búsqueda local. A diferencia de PTL, las heurísticas de búsqueda local están categorizadas en cuatro grupos: generales, de búsqueda por área, de búsqueda por región y de asignación parcial. Las heurísticas que se agregarán al algoritmo serán las que se ajusten a las características de cada problema a resolver. A continuación se listan las heurísticas.

Heurísticas de agitación:

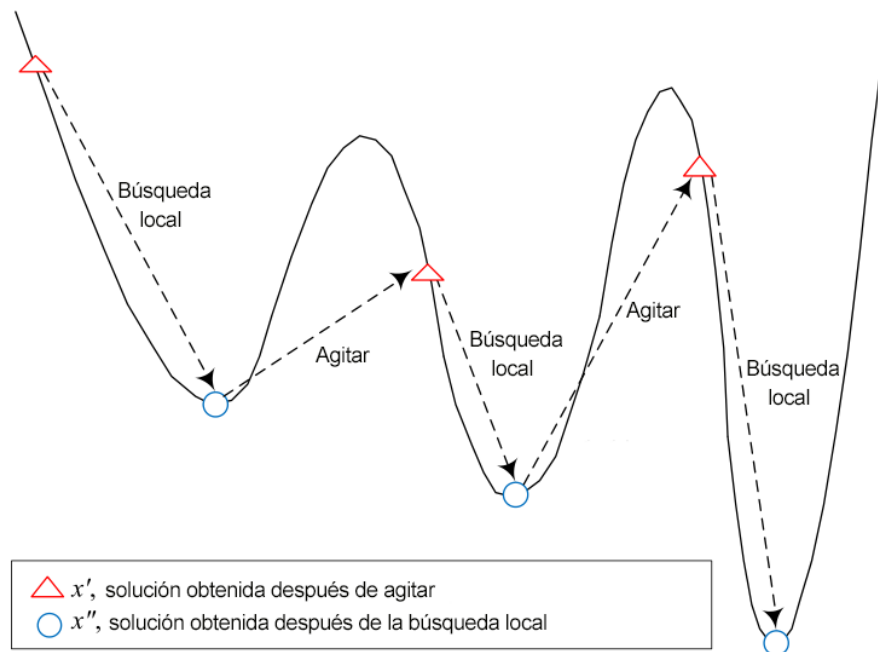


Figura 4.3: Funcionamiento básico de HH-VNS-PP. La solución se perturba para escapar del óptimo local mediante la agitación, generando x' (triángulo rojo). Luego se obtiene otro óptimo local x'' (círculo azul) mediante el paso de búsqueda local.

- **SwapRand**: Cambia estado aleatoriamente a los proyectos.
- **SwapQuarter**: Cambia estado a $1/4$ de los proyectos.
- **SwapThird**: Cambia estado a $1/3$ de los proyectos.
- **SwapHalf**: Cambia estado a $1/2$ de los proyectos.
- **ShakeArea**: Cambia estado a los proyectos de un área.
- **ShakeRegion**: Cambia estado a los proyectos de una región.
- **SwapRand**: Cambia estado aleatoriamente a los proyectos.

Heurísticas de búsqueda local:

- **Generales**
 - **Swap**: Toma un proyecto aleatorio y le cambia el estado.
 - **DrawRand**: Saca un proyecto e intenta agregar proyectos mientras el presupuesto lo permita.
 - **DrawHightBgt**: Saca el que más presupuesto requiera e intenta agregar proyectos mientras el presupuesto lo permita.
 - **DrawRPutLess**: Saca un proyecto aleatorio y agrega proyectos mientras haya presupuesto disponible comenzando por el de menor presupuesto.
 - **AddMaxBgt**: Saca proyectos aleatorios hasta que quepa el de mayor presupuesto fuera.
- **Área**
 - **SwapArea**: Quita un proyecto de un area e intenta agregar uno de otra.
 - **IncreaseBgtArea**: Quita un proyecto de un área e intenta agregar uno de la misma con mayor presupuesto.

- **DecreaseBgtArea:** Quita un proyecto de un área e intenta agregar uno de la misma con menor presupuesto.
- **Región:**
 - **SwapRegion:** Quita un proyecto de una región e intenta agregar uno de otra.
 - **IncreseBgtRegion:** Quita un proyecto de una región e intenta agregar uno de la misma con mayor presupuesto.
 - **DecreaseBgtRegion:** Quita uno de una región e intenta agregar uno de la misma con menor presupuesto.
- **Asignación parcial**
 - **DrawHightBgtFill:** Quita el proyecto que mayor presupuesto requiera e intenta agregar todos los que pueda. Cuando el presupuesto disponible no alcance para añadir otro proyecto, lo usará en los proyectos que ya están en la cartera y no están financiados al máximo, hasta que se agote.
 - **DrawRandFill:** Quita un proyecto al azar e intenta agregar todos los que pueda. Cuando el presupuesto disponible no alcance para añadir otro proyecto, lo usará en los proyectos que ya están en la cartera y no están financiados al máximo, hasta que se agote.
 - **SetPrjMinFill:** Selecciona un proyecto que esté en la cartera, lo financia al mínimo y usa el presupuesto restante para aumentar financiación de los proyectos de la cartera.
 - **UseBudget:** Si hay presupuesto disponible lo emplea para aumentar financiación.
 - **SetPrjRndFill:** Selecciona un proyecto que esté en la cartera y le asigna un presupuesto aleatorio. Si queda presupuesto disponible luego de la operación.

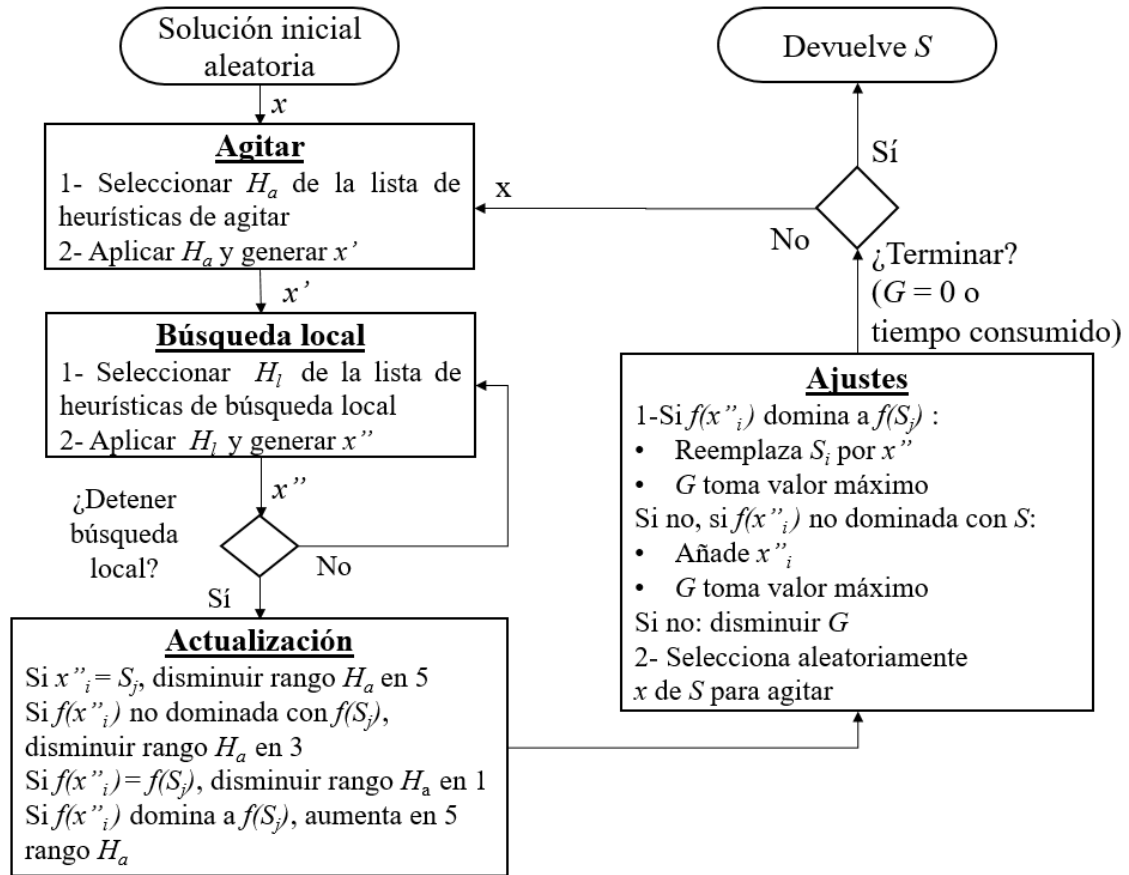


Figura 4.4: Diagrama de flujo de HH-VNS-PP.

4.2.2 ESTRUCTURA GENERAL

HH-VNS-PP consta de cuatro etapas: agitar, búsqueda local, ajustes y actualización. En la figura 4.4 se muestra el esquema general.

Una solución inicial aleatoria funge inicialmente como solución base x . Se selecciona una heurística de bajo nivel de la categoría de agitar, H_a , y se aplica a x para generar una nueva solución x' .

Las heurísticas de agitar se ordenan descendientemente de acuerdo a su rango y se elige cada vez la que mayor rango tenga. Cuando todas las heurísticas tengan rango negativo se restauran a su valor inicial.

En la etapa de búsqueda local, teniendo a x' como incumbente, se intentan

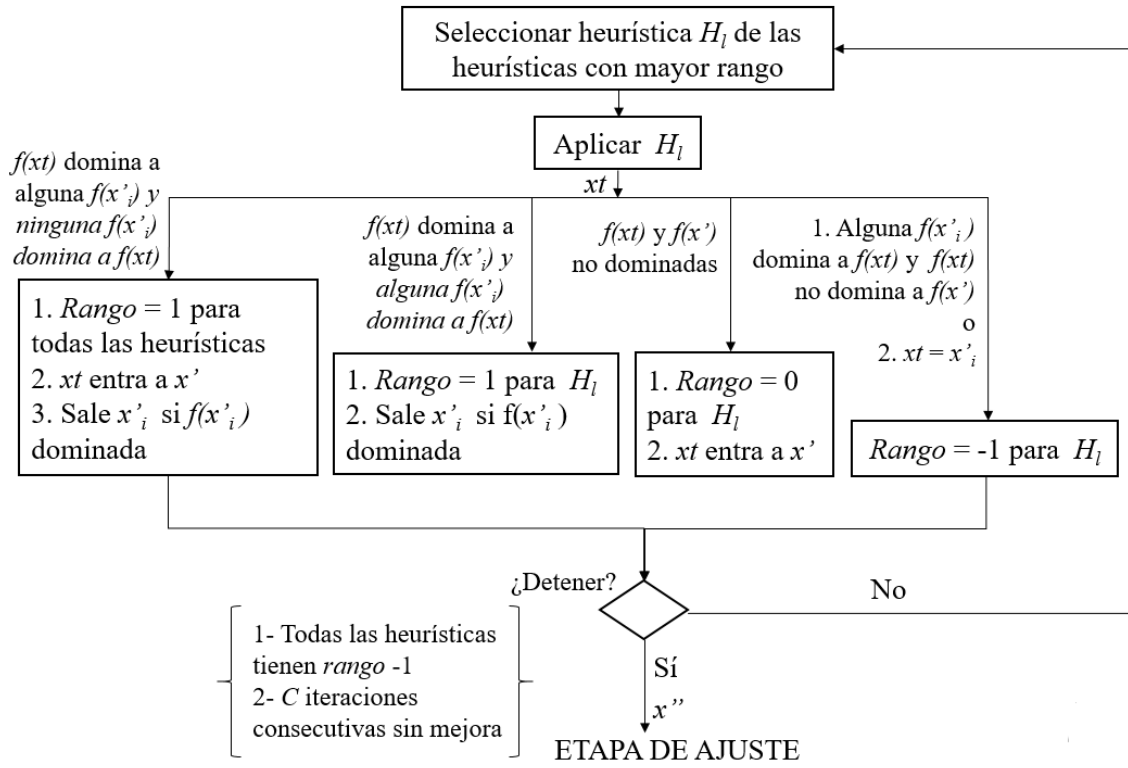


Figura 4.5: Diagrama de flujo de la etapa de búsqueda local.

encontrar las mejores soluciones en su vecindad. Inicialmente x' es una solución, a medida que la búsqueda local avanza se reemplaza por soluciones cuyas imágenes la dominen o se agregan soluciones si no hay dominancia entre ellas.

Como se observa en el diagrama de flujo de la figura 4.5, se selecciona aleatoriamente una heurística h_l entre las heurísticas de búsqueda local con mayor rango, para obtener x_t . Estas heurísticas pueden tener tres niveles o rangos: -1, 0 o 1. El valor inicial se establece en 1 y cambia de la siguiente manera:

- Si $f(x_t)$ domina a alguna de las $f(x')$ (imágenes de las soluciones que se tienen como mejores soluciones dentro de la búsqueda local) y no es dominada por ninguna, todas las heurísticas reciben rango 1. La nueva solución es agregada a x' y se sacan las soluciones cuyas imágenes fueron dominadas.
- Si $f(x_t)$ domina a alguna de las $f(x')$ y es a la vez dominada por alguna, las que son dominadas por ella se sacan de x' , h_l recibe rango 1.

- Si $f(x_t)$ es no dominada respecto a todas las $f(x')$, se añade a x' . h_l recibe rango 0.
- Si $f(x_t)$ no domina a ninguna $f(x')$ y es dominada por alguna, o x_t es igual a alguna x' , h_l recibe rango -1.

La etapa de ajuste lleva el control de las heurísticas de agitar. De acuerdo a las soluciones obtenidas de la etapa de búsqueda local, el rango de H_a es modificado de la siguiente forma:

- Si alguna de las soluciones de x'' es igual a alguna de las soluciones de S (conjunto de soluciones globales), el rango de H_a disminuye en 10.
- Por cada $f(x''_i)$ que sea dominada por alguna $f(S_j)$, se disminuye el rango de H_a en 5.
- Por cada $f(x''_i)$ que sea no dominada con alguna $f(S_j)$, se disminuye el rango de H_a en 1.
- Por cada $f(x''_i)$ que domine a alguna $f(S_j)$, se aumenta el rango de H_a en 1.

En la etapa de actualización se evalúan las soluciones obtenidas de la búsqueda local y se actualiza el conjunto de soluciones eficientes globales S . Además, se ajusta el parámetro G , que representa el número de iteraciones consecutivas sin mejora del algoritmo en general.

La actualización realiza como sigue:

- Si alguna de las soluciones de x'' es igual a algunas de las soluciones de S (conjunto de soluciones globales), $f(h_a)$ es agregada a la lista tabu.
- Si alguna $f(x''_i)$, domina a $f(S_j)$:
 - Se añade x''_i a S .

- Se extraen de S todas las soluciones cuyas imágenes fueron dominadas.
- G adquiere su máximo valor.
- De no ocurrir lo anterior, si alguna $f(x_i'')$ es no dominada con todas las $f(S_j)$:
 - Se añade x_i'' a S
 - G toma máximo valor.
- De no ocurrir ninguna de las anteriores:
 - Disminuir G en una unidad.

Durante este proceso se extraen de S todas las soluciones cuyas imágenes hayan sido dominadas, por lo que siempre va a contener un conjunto de soluciones eficientes.

Para iniciar la siguiente iteración se selecciona aleatoriamente una de esas soluciones. Todo el algoritmo se repite hasta que G tenga valor cero o se consuma el tiempo establecido.

4.3 METODOLOGÍA DE APOYO A LA DECISIÓN

La metodología de apoyo a la decisión propuesta para abordar el problema de selección de cartera de proyectos incorpora las siguientes etapas:

1. Identificación de las características del problema
2. Definir las preferencias del tomador de decisiones
3. Ajuste de datos de entrada
4. Generación del conjunto de carteras
5. Presentación del conjunto de carteras al tomador de decisiones

4.3.1 IDENTIFICACIÓN DE LAS CARACTERÍSTICAS DEL PROBLEMA

Esta etapa permite determinar si HH-VNS-PP puede ser empleado en el problema en cuestión. Para ello se requiere adquirir información detallada del problema.

Lo primero a tener en cuenta es la vigencia en la cartera de los proyectos. Si la cartera es dinámica, no podrá ser utilizado HH-VNS-PP.

Posterior se deben evaluar cada una de las características, intentando encontrar alguna que invalide el uso de HH-VNS-PP. Si se encuentra alguna, es necesario buscar otro método de resolución al problema pues esta metodología no podrá ser empleada. De no encontrar ningún impedimento, se pasa a la siguiente etapa.

4.3.2 DEFINIR LAS PREFERENCIAS DEL TOMADOR DE DECISIONES

En esta etapa se definen las preferencias del tomador de decisiones sobre cada objetivo para asignar pesos a cada uno de ellos. Si el tomador de decisiones no desea definir preferencias, a cada objetivo se le asigna el mismo peso.

El método de decisión multicriterio que se emplea en esta metodología es el ELECTRE, debido a la simplicidad que ofrece al momento de definir los pesos de los objetivos. No obstante, como en nuestra metodología el método de decisión multicriterio se emplea a posteriori, pudiese emplearse otro, si el tomador de decisiones así lo desea. La metodología de apoyo a la decisión que se propone es flexible en ese aspecto.

El decisor define además cuántas soluciones localmente eficientes desea recibir para, sobre ellas, hacer una elección final.

4.3.3 AJUSTE DE DATOS DE ENTRADA

En esta etapa se ajustan los datos la(s) instancia(s) del problema de modo que se ajuste a los requerimientos de entrada HH-VNS-PP, descritos a continuación:

1. Cantidad de proyectos.
2. Presupuesto disponible.
3. Definir si la asignación es total o parcial. Si es parcial, definir el porcentaje de beneficio que brindará el proyecto en caso de ser financiado con el mínimo permitido.
4. Cantidad de objetivos.
5. Pesos de los objetivos (de haberlos).
6. Cantidad de estructuras en la que se organiza (regiones, área, otras)
7. Definir mínimo y máximo de presupuesto a utilizar en cada estructura de organización (de haberlas).
8. Cantidad de actividades o tareas (1 si la estructura de proyectos es única).
9. Cantidad de sinergias (0 si no hay)

A continuación se presentan todos los proyectos siguiendo la estructura:

1. Identificador del proyecto. De no tenerlo, se toma la posición en la lista de proyectos.
2. Si la asignación es total, financiamiento requerido. Si la asignación es parcial, financiamiento mínimo necesario y financiamiento máximo.
3. Las estructuras organizacionales a las que pertenece. Ejemplo: [Área 4, Región 8].

4. Beneficio en los diferentes objetivos.

Si los proyectos están divididos en actividades o tareas, para cada proyecto habrá una lista de las mismas con la siguiente estructura:

1. Identificador de la actividad
2. Porcentaje del beneficio del proyecto que aporta la actividad.
3. Financiamiento mínimo necesario para llevar a cabo la actividad.
4. Financiamiento para que la actividad aporte su máximo beneficio.

De existir sinergias, se listan siguiendo la estructura:

1. Identificación de la sinergia
2. Tipo de sinergia (aumento de beneficio, disminución de beneficio, aumento de costo, disminución de costo, incompatibilidad)
3. Valor de la sinergia
4. Cantidad de elementos en la sinergia
5. Mínimo para activar
6. Máximo para activar

Para cada sinergia, se listan los elementos que la componen de la forma:

1. Identificador de la sinergia a la que pertenece
2. Identificador del proyecto
3. Identificador de la tarea o actividad

4.3.4 GENERACIÓN DEL CONJUNTO DE CARTERAS

HH-VNS-PP posee tres parámetros que se pueden ajustar para generar el conjunto de carteras.

1. Número de soluciones dominadas consecutivas encontradas en la búsqueda local para salir de la iteración: Está ajustado por defecto en 20. Un valor más pequeño hace más rápida cada iteración pero podrían perderse mejores soluciones. Un valor mayor podría ofrecer mejores soluciones pero ralentiza el proceso al iterar más veces sobre zonas en las que no se encuentran buenas soluciones.
2. Número iteraciones de búsqueda local consecutivas sin cambios en la lista de soluciones localmente eficientes para terminar el algoritmo y devolver las soluciones encontradas hasta entonces. Por defecto este valor es 5.
3. Tiempo máximo de ejecución: Si el tiempo de ejecución al terminar la búsqueda local es mayor que este parámetro, HH-VNS-PP devuelve las soluciones encontradas hasta el momento.
4. Las soluciones son puntuadas mediante el método ELECTRE (u otro método si el tomador de decisiones así lo prefiera) y ordenadas según esa puntuación.

4.3.5 PRESENTACIÓN DEL CONJUNTO DE CARTERAS AL TOMADOR DE DECISIONES Y SELECCIÓN FINAL

Se le presentan al tomador de decisiones el número máximo de posibles soluciones que pidió. Este puede elegir una de las ofrecidas o decidir que desea hacer algún cambio en alguna de las etapas de la metodología, esperando obtener resultados diferentes. Este proceso puede repetirse hasta que el tomador de decisiones obtenga una solución con la cual se sienta satisfecho.

CAPÍTULO 5

EXPERIMENTACIÓN

En este capítulo se describen las instancias empleadas en el proceso de experimentación. Para todas las instancias se encontraron soluciones factibles, algunas de las cuales son mostradas y analizadas.

El proceso de experimentación fue realizada en una MacBook Air con procesador Intel Core i7-4650U a 1.70 GHz con 8 GB de memoria RAM. El sistema operativo empleado fue Windows 10 de 64 bits.

El algoritmo fue programado en Python 3 empleando Spyder 3, de Anaconda 4. El análisis y visualización de los resultados fue realizado con las librerías `pandas`, `matplotlib` `seaborn`, en el entorno de Jupyter Notebook.

Para el proceso de experimentación se emplean instancias con características diferentes, tomadas de varias fuentes. Para facilitar la comprensión se organizan en tres grupos. Resúmenes de las mismas son mostrados en sus secciones correspondientes en cuadros con la siguiente estructura:

- *CI*: Cantidad de instancias con esas características que se tienen
- *P*: Cantidad de proyectos en la instancia
- *O*: Cantidad de objetivos

- A : Cantidad de áreas
- R : Cantidad de regiones
- AT : Asignación total
- AP : Asignación parcial
- T : Tareas
- S : Sinergias

5.1 GRUPO A

Las instancias del Grupo A, cuyas características muestra el cuadro 5.1, han sido trabajadas por otros autores anteriormente [3, 4, 46] . La mayor complejidad radica en la granularidad de los proyectos en tareas y las sinergias entre ellas, lo que hace que con cada cambio que se haga en la cartera, haya que chequear la existencia de las mismas y actualizar los valores asociados, lo que hace considerablemente más lenta la búsqueda.

En este grupo se desea maximizar el número de proyectos en la cartera y el beneficio total obtenido. Cada proyecto posee un requerimiento mínimo y máximo de recursos y cada proyecto está granulado en tareas que, a su vez, solicitan un mínimo y máximo de recursos. La manera en la que se maneja el beneficio obtenido por cada tarea en los proyectos, varía en la bibliografía. En este trabajo se asume que si la tarea recibe el máximo de recursos, proporcionará el máximo beneficio; mientras que si recibe el mínimo, proporcionará el 70 % del beneficio, cualquier financiación intermedia obtendrá un beneficio que es una función lineal de estos valores.

La literatura trabaja estas instancias con modelos exactos y desde diferentes ángulos. De manera general, la función objetivo es representada a partir de la propuesta por Litvinchev [47], donde el valor del parámetro α tiene un rol fundamental.

Cuadro 5.1: Características de las instancias del Grupo A [3].

CI	P	O	A	R	AT	AP	T	S
4	64	2	8	No	No	Sí	16	2
4	64	2	12	No	No	Sí	16	No
15	100	2	2	No	No	Sí	5	5
15	100	2	2	No	No	Sí	5	No
4	256	2	8	No	No	Sí	16	10
4	256	2	8	No	No	Sí	16	10
15	500	2	2	No	No	Sí	15	40
15	500	2	2	No	No	Sí	15	No
4	512	2	8	No	No	Sí	16	No
4	512	2	8	No	No	Sí	16	20
5	1024	2	8	No	No	Sí	16	40
5	1024	2	8	No	No	Sí	32	40

Mediante el contacto directo con Arratia [3], fue posible tener acceso a algunos resultados de su experimentación. No obstante, en la mayoría no se especifica el parámetro α empleado, lo que imposibilita una comparación de resultados.

En las 15 instancias de 100 proyectos sin sinergias se conoce que el valor α es de 0,8. En su caso los objetivos son ponderados con igual peso para obtener un valor objetivo único, mientras que en el presente se obtiene un conjunto de soluciones eficientes.

En la figura 5.3 se muestran los valores encontrados por HH-VNS-PP en color azul y en rojo el valor obtenido de la literatura. Como puede apreciarse, aunque las instancias tienen la misma cantidad de proyectos, las características de cada una individualmente influyen en la calidad y cantidad de las soluciones obtenidas.

Aunque la aleatoricidad del algoritmo y el tiempo de ejecución es un factor determinante, los valores mostrados para este grupo de instancias son el resultado

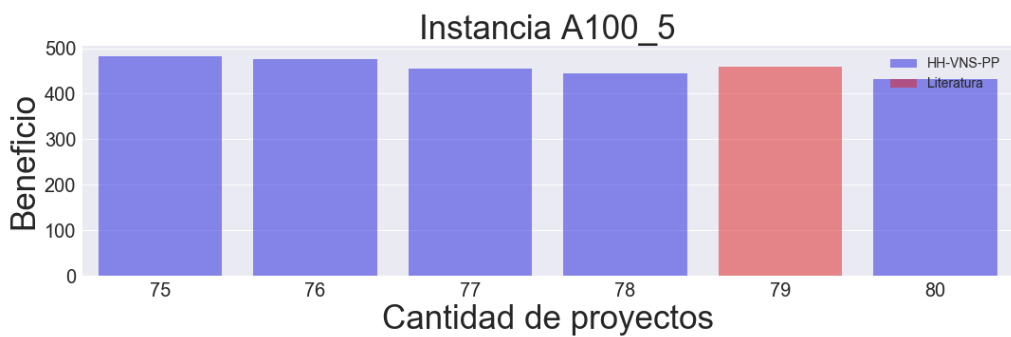
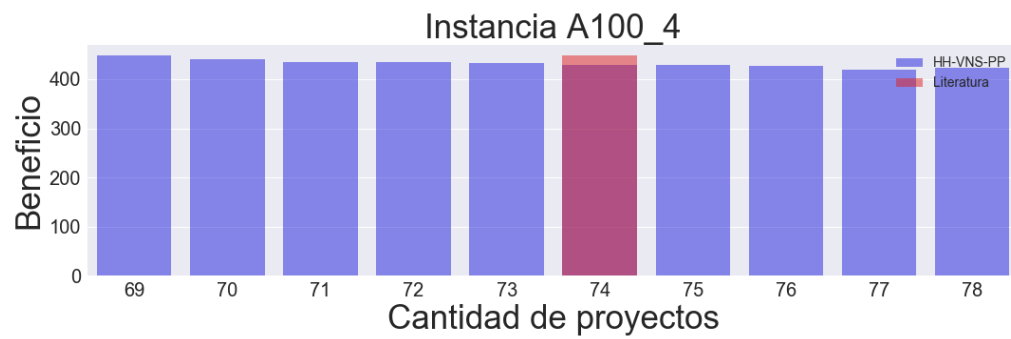
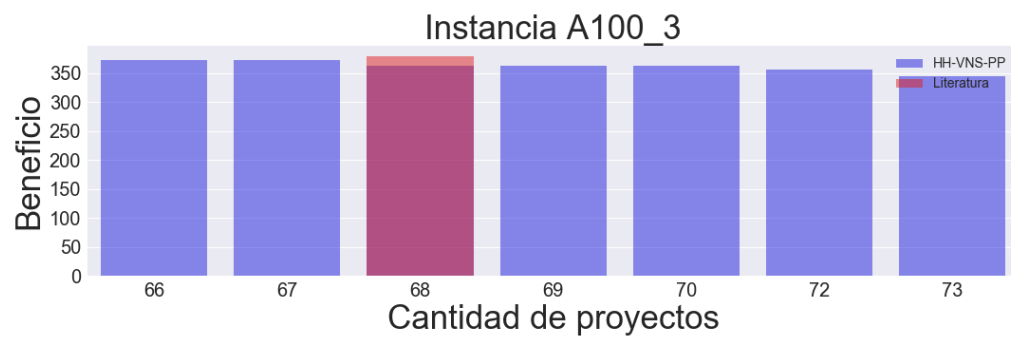
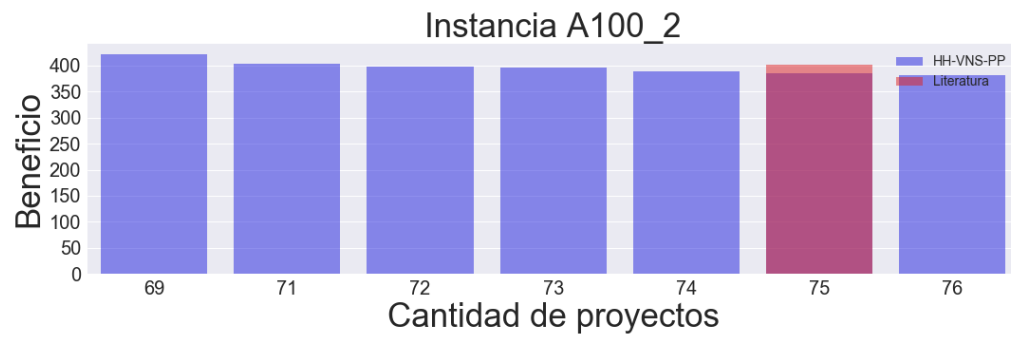
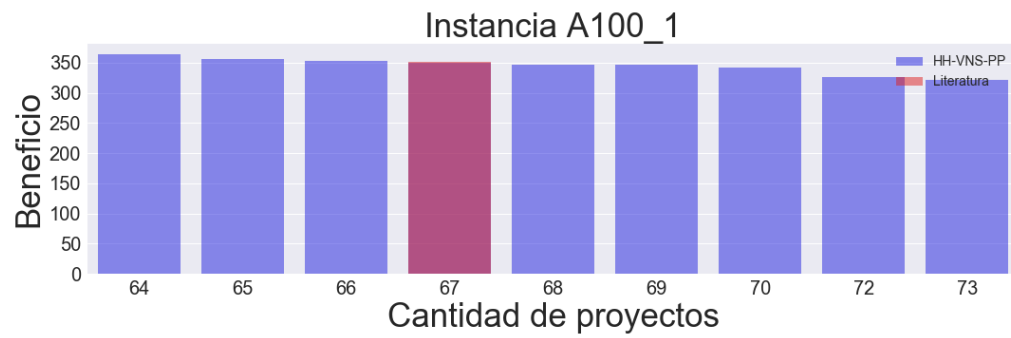
de una sola ejecución, con una duración máxima de 1 hora. El objetivo de esto es mostrar algunas características específicas que pueden tener las soluciones en particular.

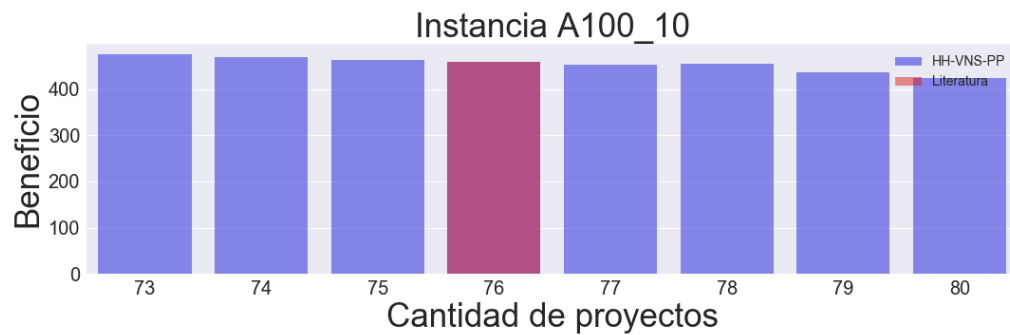
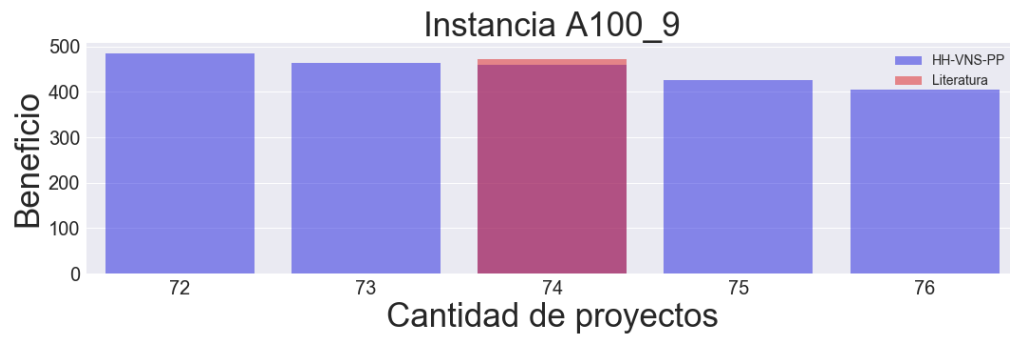
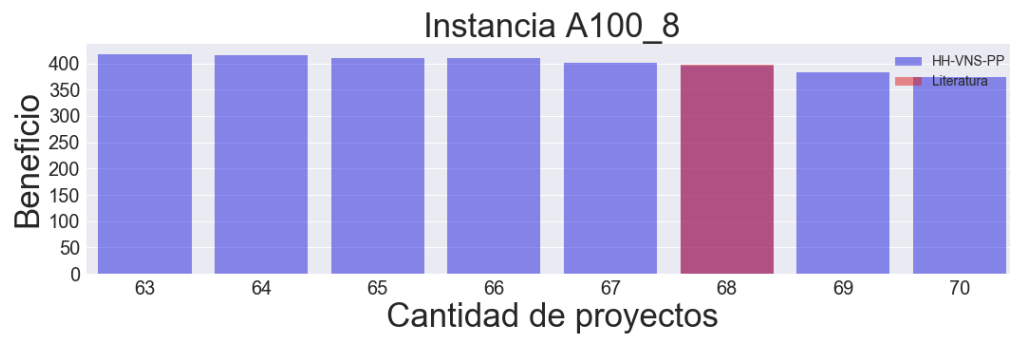
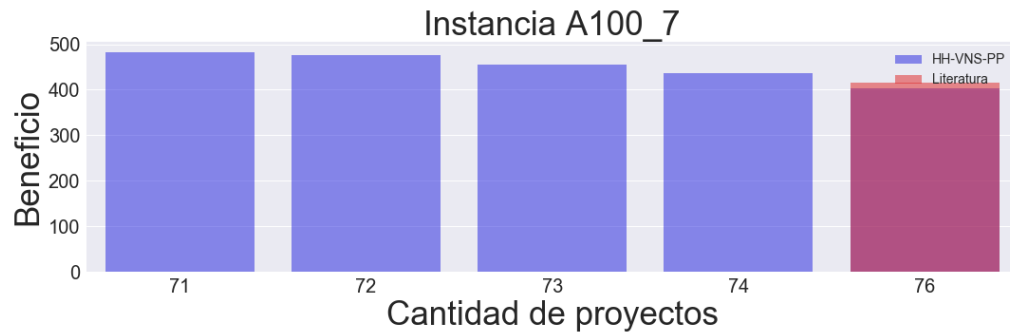
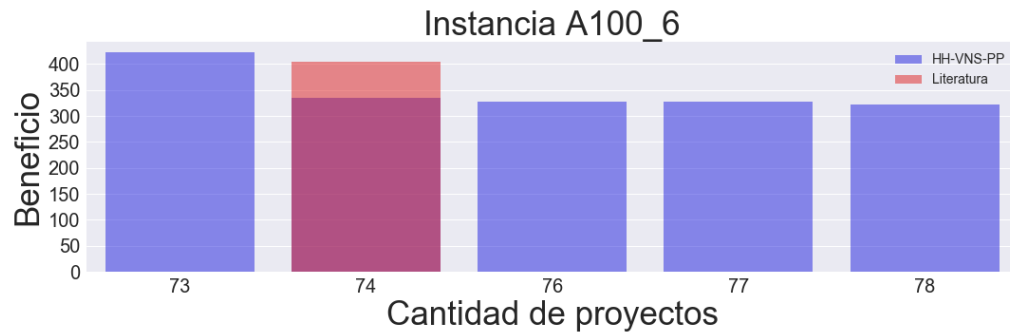
En el cuadro 5.2 se muestra un resumen de resultados obtenidos respecto a las soluciones exactas que se tienen de referencia. La información está organizada de la siguiente manera:

- **PL:** Número de proyectos en la solución óptima de la literatura.
- **BL:** Beneficio de la solución de la literatura.
- **CS:** Cantidad de soluciones eficientes obtenidas por HH-VNS-PP.
- **SMCP:** Cantidad de soluciones en las que hay más proyectos que en la literatura.
- **CSMB:** Cantidad de soluciones en las que el beneficio es mayor que en la solución de la literatura.
- **BSICP:** Beneficio de la solución obtenida para igual cantidad de proyectos que en la solución de la literatura.
- **Porc:** Porcentaje de beneficio obtenido para el mismo número de proyectos que en la solución de la literatura.

El promedio del beneficio obtenido por HH-VNS-PP para la misma cantidad de proyectos del óptimo en la referencia es del 96,1. En la instancia A100_11 se alcanzó el óptimo mientras que el menor valor porcentual se obtuvo en la instancia A100_6, donde el beneficio fue del 83 % del óptimo para 74 proyectos, aunque en tres de las cinco carteras encontradas se incluyen más proyectos y en una se obtiene mayor beneficio.

En la instancia A100_5 ocurre algo particular. Aunque HH-VNS-PP devuelve cinco carteras, ninguna de ellas tiene la misma cantidad de proyectos que la de





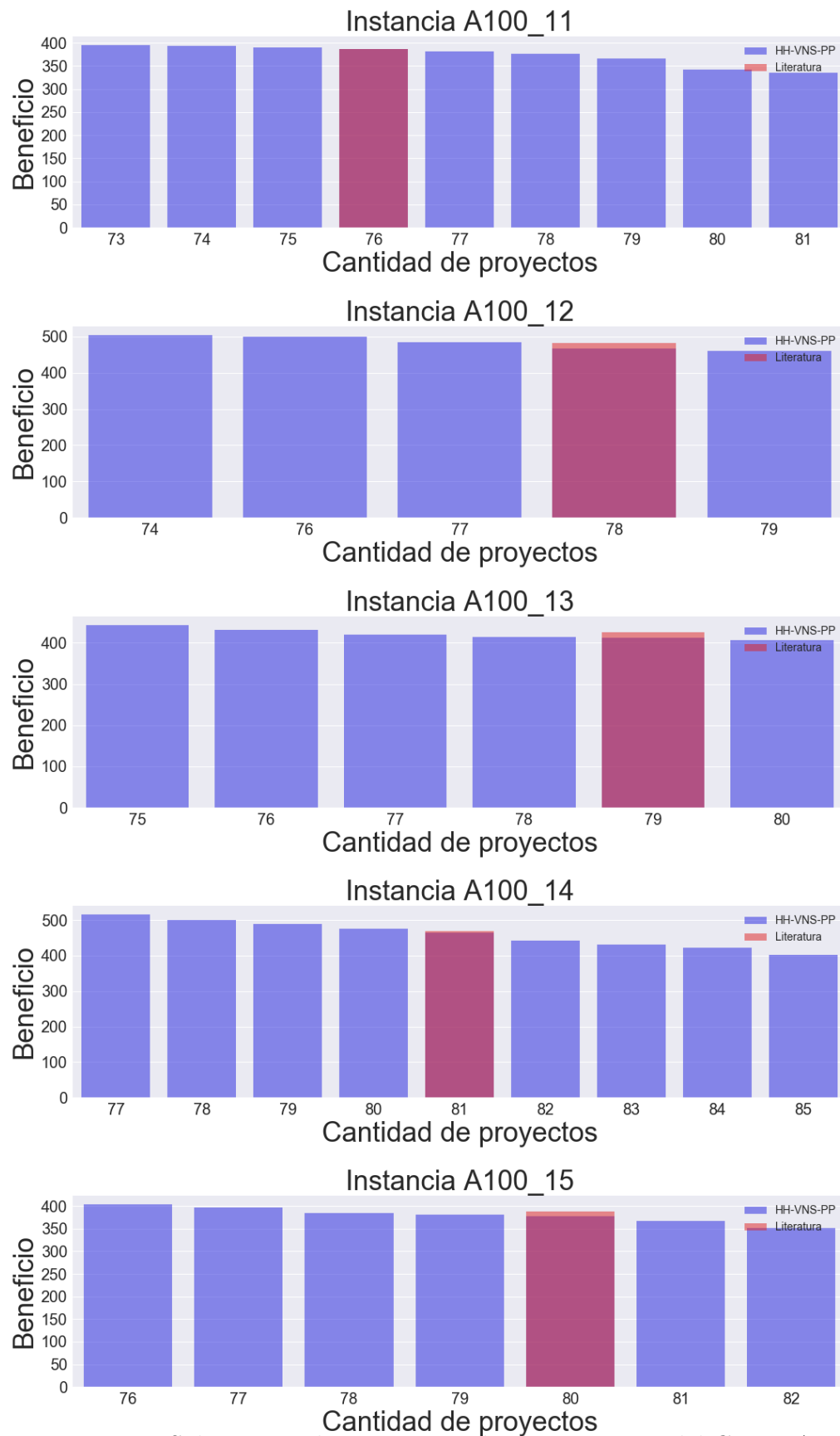


Figura 5.3: Soluciones a las instancias de 100 proyectos del Grupo A.

la referencia, por lo tanto, no se pueden comparar esos valores. En una de ellas se incluyen más proyectos y en dos se alcanza mayor beneficio.

Cuadro 5.2: Resumen de resultados para las instancias A100.

	En literatura		HH-VNS-PP				
	PL	BL	CS	SMCP	CSMB	BSICP	Porc
A100_1	67	351,5	9	5	3	348,9	99,2
A100_2	75	410,9	7	1	1	384,8	93,6
A100_3	68	378,2	7	4	0	363,1	96,0
A100_4	74	447,8	10	4	0	430,0	96,0
A100_5	79	458,8	5	1	2	-	-
A100_6	74	403,9	5	3	1	335,3	83,0
A100_7	76	416,0	5	0	4	386,9	96,9
A100_8	68	396,7	8	2	5	396,1	99,8
A100_9	74	472,1	5	1	2	460,3	97,5
A100_10	76	459,4	8	4	3	458,9	99,9
A100_11	76	386,1	9	5	3	386,1	100,0
A100_12	78	482,3	5	2	1	467,5	96,9
A100_13	79	426,0	6	1	2	412,8	96,9
A100_14	81	469,1	9	4	4	464,0	98,9
A100_15	80	388,3	7	2	2	377,2	97,1

En la figura 5.4 se muestra el desempeño de las heurísticas de búsqueda local para este grupo de instancias. **DrawHightBgtFill** fue empleada 5 765 680 veces, un 30 % de veces más que la que le sigue con 4 420 040 usos, **SetMinFill**. El resto de las heurísticas tuvieron un desempeño similar y la diferencia entre los usos nunca excede el 20 %.

En las heurísticas de agitar **SwapQuarter** destaca sobre **SwapThird** con una diferencia de apenas el 11 %. El desempeño de **AllRand** y **ShakeArea** fue muy bajo respecto al resto de las heurísticas para estas instancias.

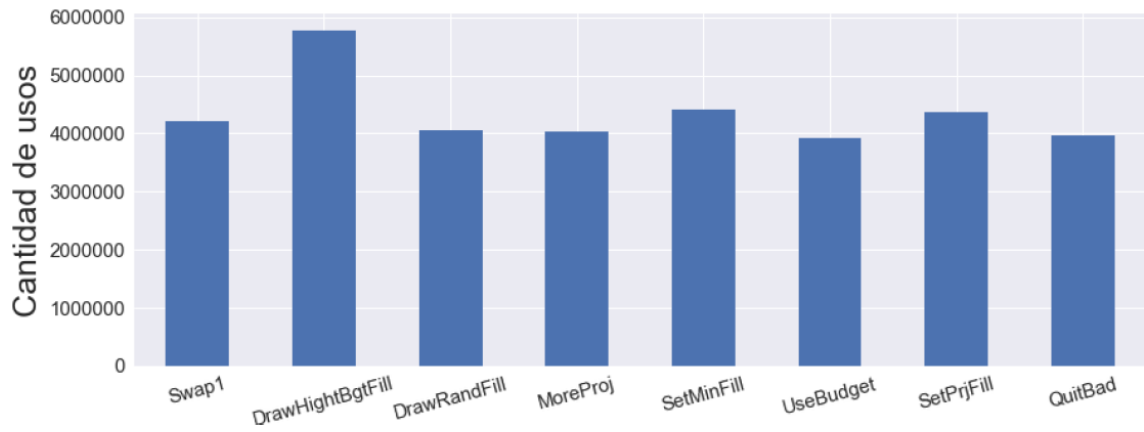


Figura 5.4: Desempeño de heurísticas de búsqueda local para el Grupo A.

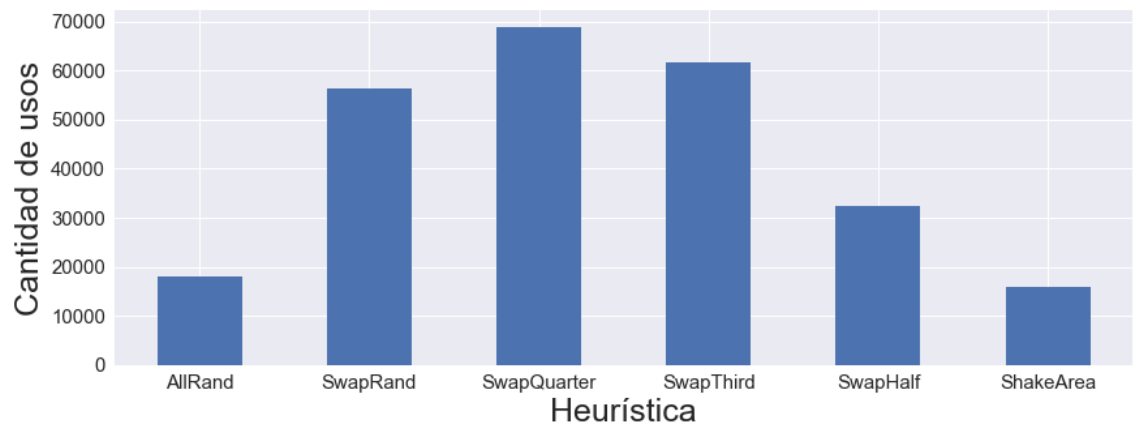


Figura 5.5: Desempeño de heurísticas de agitar para el Grupo A.

5.2 GRUPO B

En las instancias del Grupo B (cuadro 5.3), obtenidas de Galván y García [31, 32], la mayor dificultad se encuentra en el número de objetivos, que pueden ser 4 o 9. Los autores realizan una suma ponderada de los objetivos, obteniendo solo un valor objetivo óptimo, mientras que el presente trabajo maneja los objetivos de manera independiente, obteniendo un conjunto de soluciones eficientes y aplicando a posterior las preferencias del tomador de decisiones.

Cuadro 5.3: Características de instancias del Grupo B [31, 32].

CI	P	O	A	R	AT	AP	T	S
3	30	9	3	2	Sí	No	No	No
3	20	4	3	2	Sí	No	No	No
3	20	9	3	2	Sí	No	No	No
5	100	9	3	2	Sí	No	No	No

Como se observa en el cuadro 5.3, hay cuatro tipos de instancias en el Grupo B, en las que varían la cantidad de proyectos y objetivos. Al ejecutar HH-VNS-PP con diferentes tiempos, se puede apreciar el comportamiento del algoritmo.

En las instancias de 20 proyectos y cuatro objetivos, la cantidad de soluciones eficientes encontradas para los diferentes tiempos de ejecución no ofrece grandes variaciones, como evidencia la figura 5.6.

Aunque no es un medidor específico de calidad en las soluciones, el beneficio máximo alcanzado para cada objetivo puede dar una idea del comportamiento del algoritmo. Al comparar este valor en los diferentes tiempos de ejecución, se aprecia que hasta las instancias de 100 proyectos y nueve objetivos no se observan grandes variaciones, como se muestra en las figuras 5.7, 5.8, 5.9 y 5.10.

En las instancias de 20 proyectos y 9 objetivos la tendencia es a que la cantidad

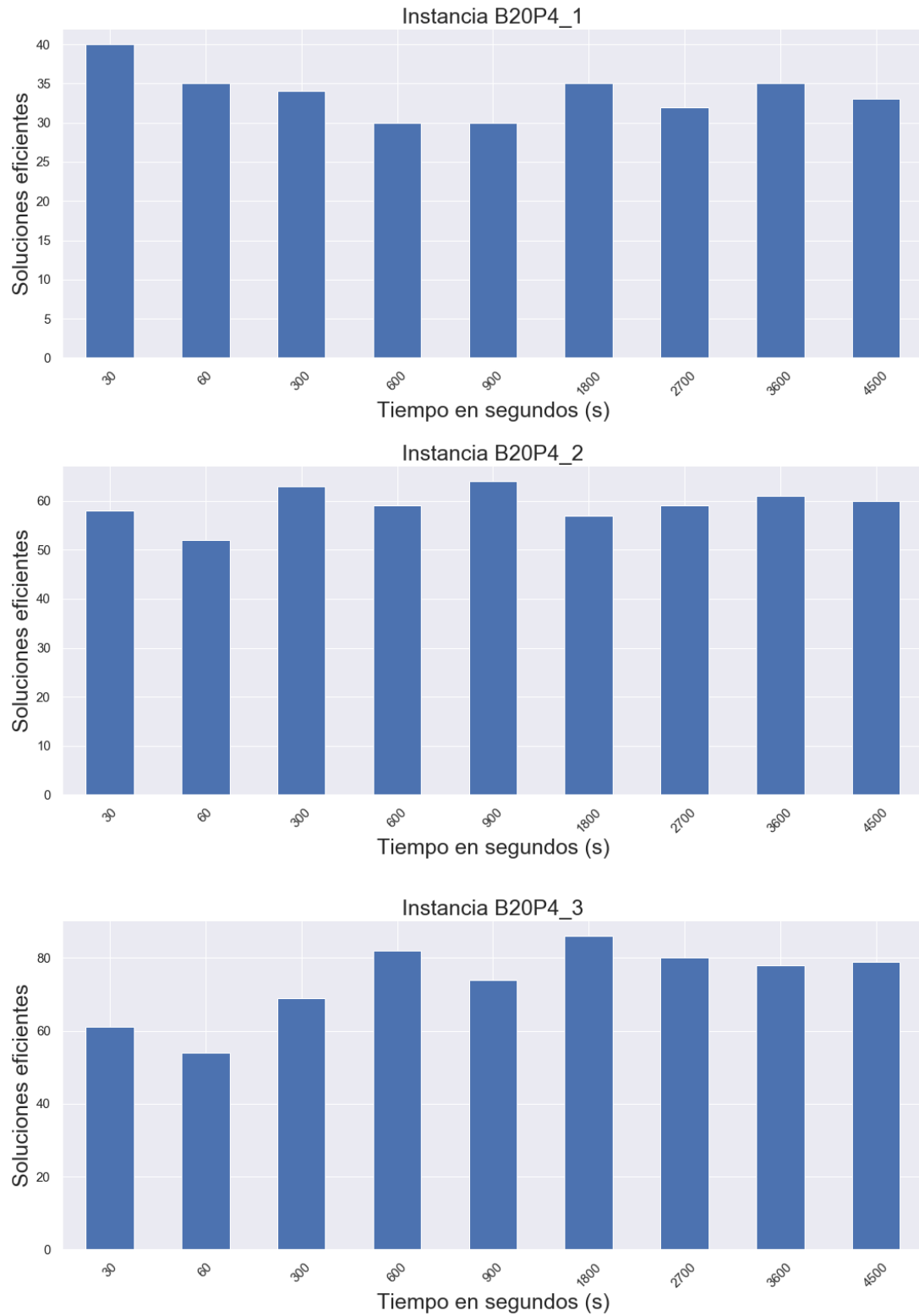


Figura 5.6: Cantidad de soluciones eficientes encontradas en instancias de 20 proyectos y cuatro objetivos en el Grupo B.

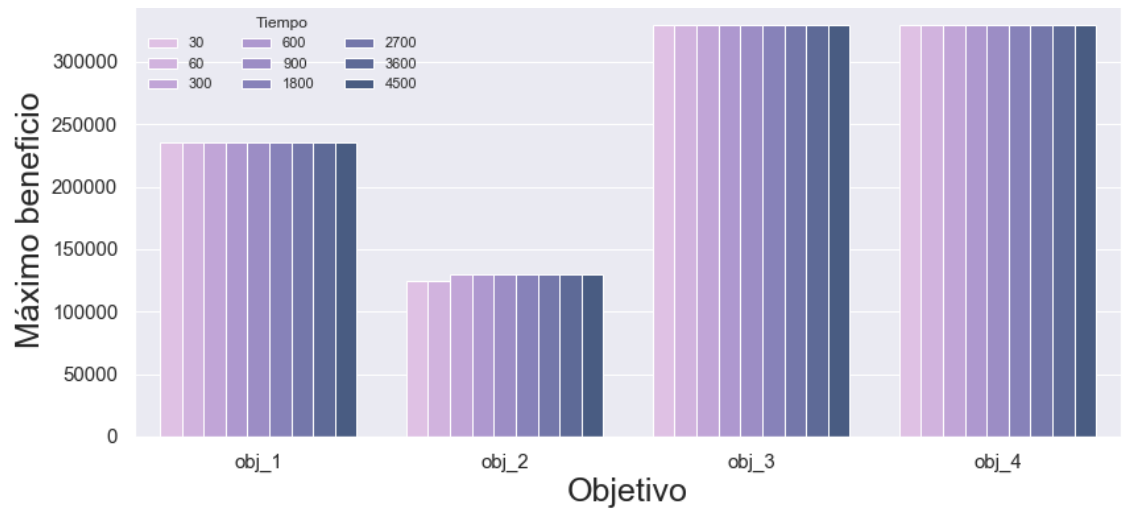


Figura 5.7: Beneficio máximo para cada objetivo en diferentes tiempos, instancia B20P4.1.

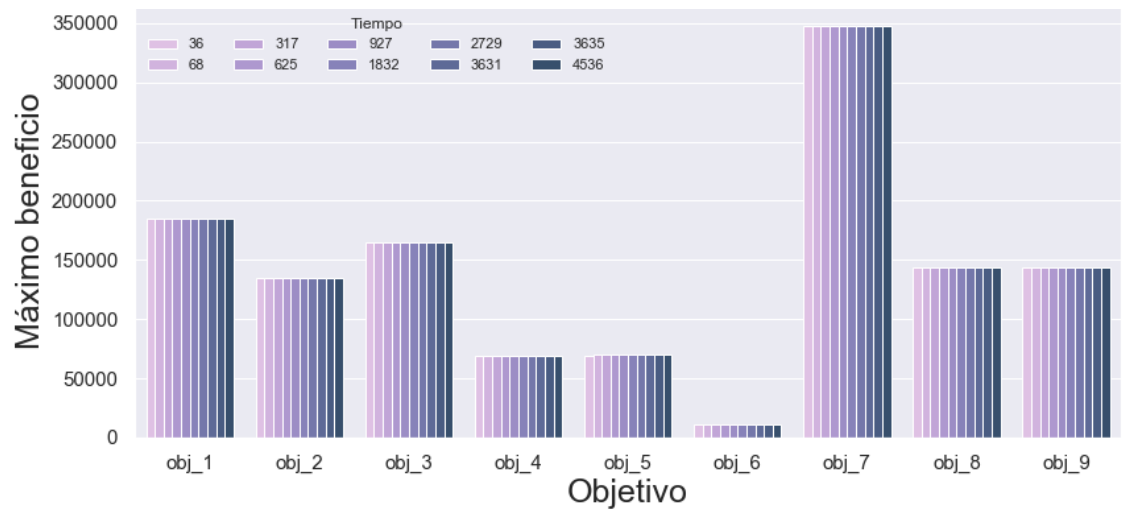


Figura 5.8: Beneficio máximo para cada objetivo en diferentes tiempos, instancia B20P9.1.

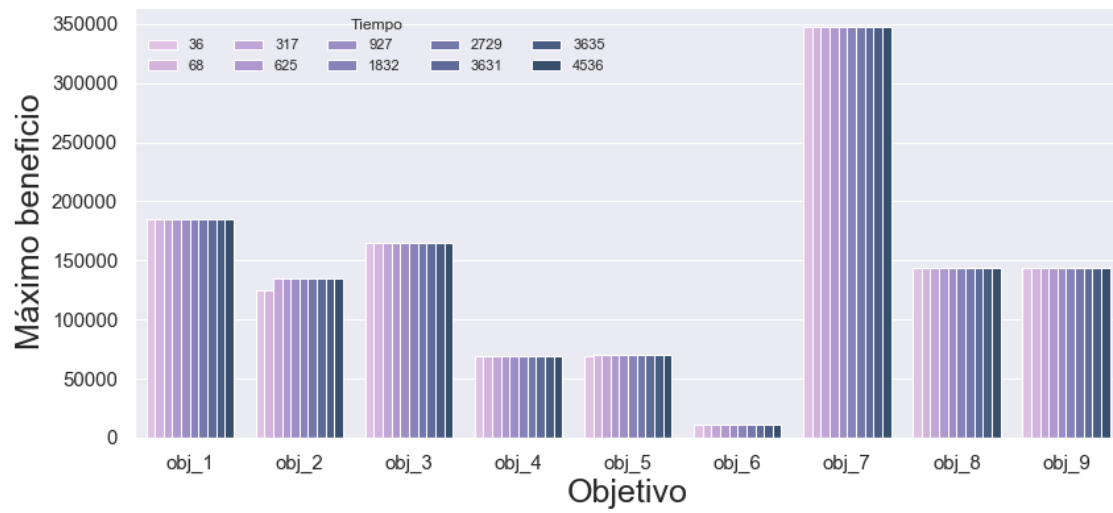


Figura 5.9: Beneficio máximo para cada objetivo en diferentes tiempos, instancia B30P9.1.

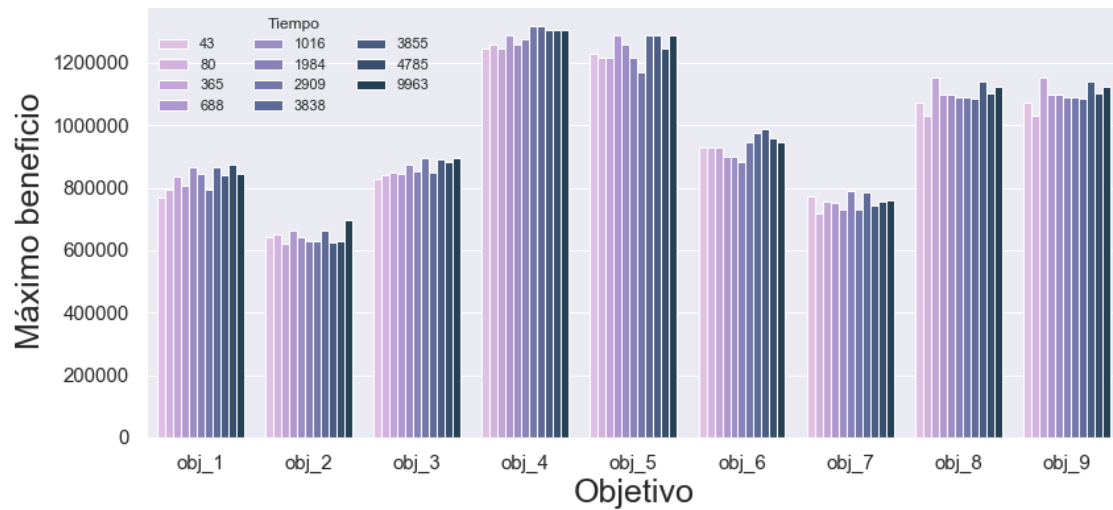


Figura 5.10: Beneficio máximo para cada objetivo en diferentes tiempos, instancia B100P9.1.

de soluciones eficientes encontradas aumente hasta un tiempo que varía para cada instancia, partir del cual la cantidad de soluciones eficientes encontradas es muy similar. Esto se observa claramente en la figura 5.11.

En la figura 5.14 se muestra la cantidad de soluciones eficientes que para las instancias de 30 proyectos y nueve objetivos. En las instancias 1 y 2 observa que luego de aproximadamente 1 hora, la cantidad de soluciones eficientes encontradas se mantiene estable. En la instancia 3, para dos horas se sigue incrementando el número de soluciones encontradas.

Las heurísticas empleadas para agitar en este grupo fueron **SwapRandom**, **SwapQuarter**, **SwapThird**, **SwapHalf**, **ShakeArea** y **ShakeRegion**. Como puede apreciarse en la figura 5.12, al igual que en el Grupo A se emplearon seis heurísticas, con la diferencia de que en este se usó **ShakeRegion** en lugar de **AllRand**. Nuevamente, la heurística que más se usó y por tanto, la que mejores resultados arrojó fue **SwapQuarter**.

Para la búsqueda local se emplearon 13 heurísticas: **Swap1**, **DrawHightBgt**, **AddRand**, **AddLowBgt**, **DrawRand**, **DrawHightBgt**, **DrawRPutLess**, **AddMaxBgt**, **SwapArea**, **IncBgtArea**, **DecBgtArea**, **SwapRegion** e **IncBgtRegion**. Como se observa en la figura 5.13, las que mejores rendimientos tuvieron fueron **SwapArea**, **SwapRegion** y **DrawRPutLess**.

5.3 GRUPO C

Para aumentar la diversidad, el Grupo C de instancias fue generado aleatoriamente. Estas instancias poseen asignación parcial como las del Grupo A y al menos cuatro objetivos, como las del Grupo B, como se observa en el cuadro 5.4.

Estas instancias fueron creadas aleatoriamente con el fin de observar la influencia de las sinergias en el desempeño del algoritmo, particularmente en el número de soluciones eficientes encontradas en el tiempo. Esto es posible debido a que estas ins-

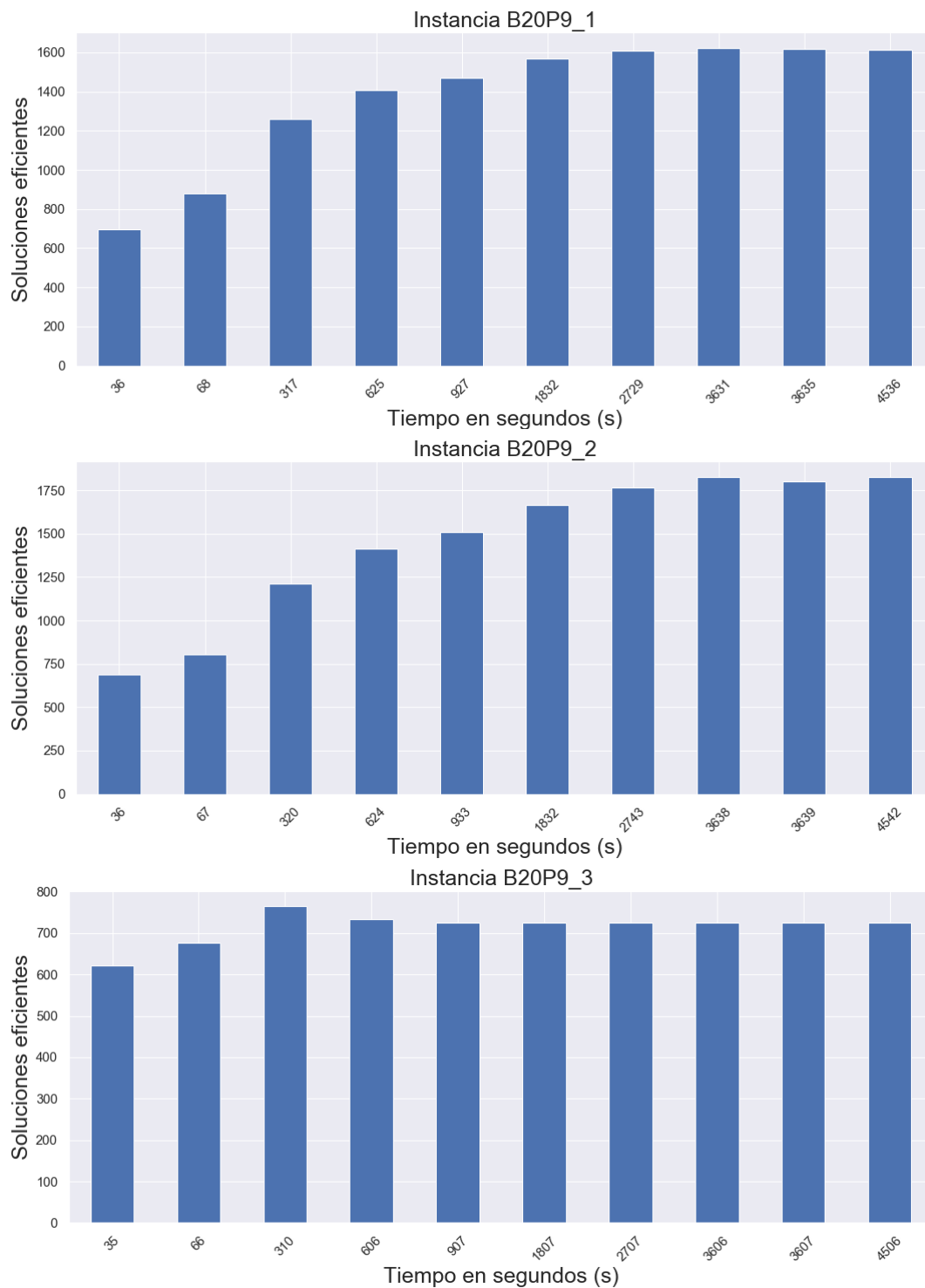


Figura 5.11: Cantidad de soluciones eficientes encontradas en instancias de 20 proyectos y nueve objetivos en el Grupo B.

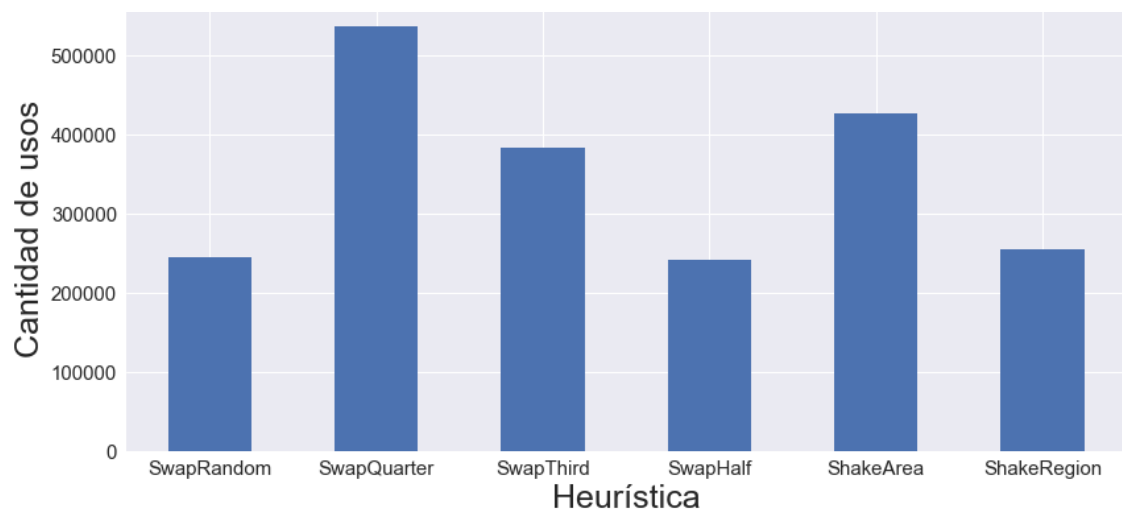


Figura 5.12: Desempeño de heurísticas de agitar para el Grupo B.

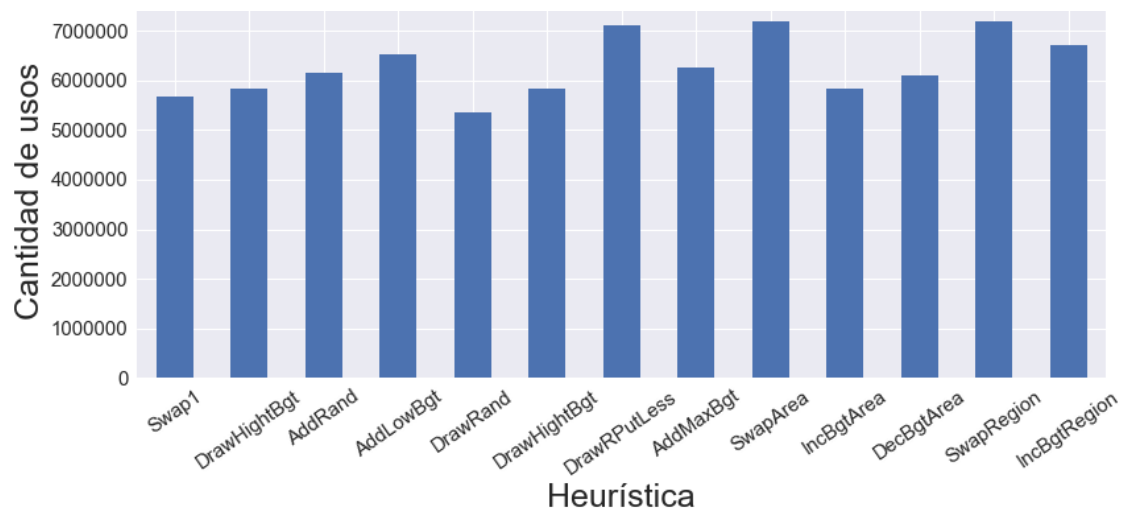


Figura 5.13: Desempeño de heurísticas de búsqueda local para el Grupo B.

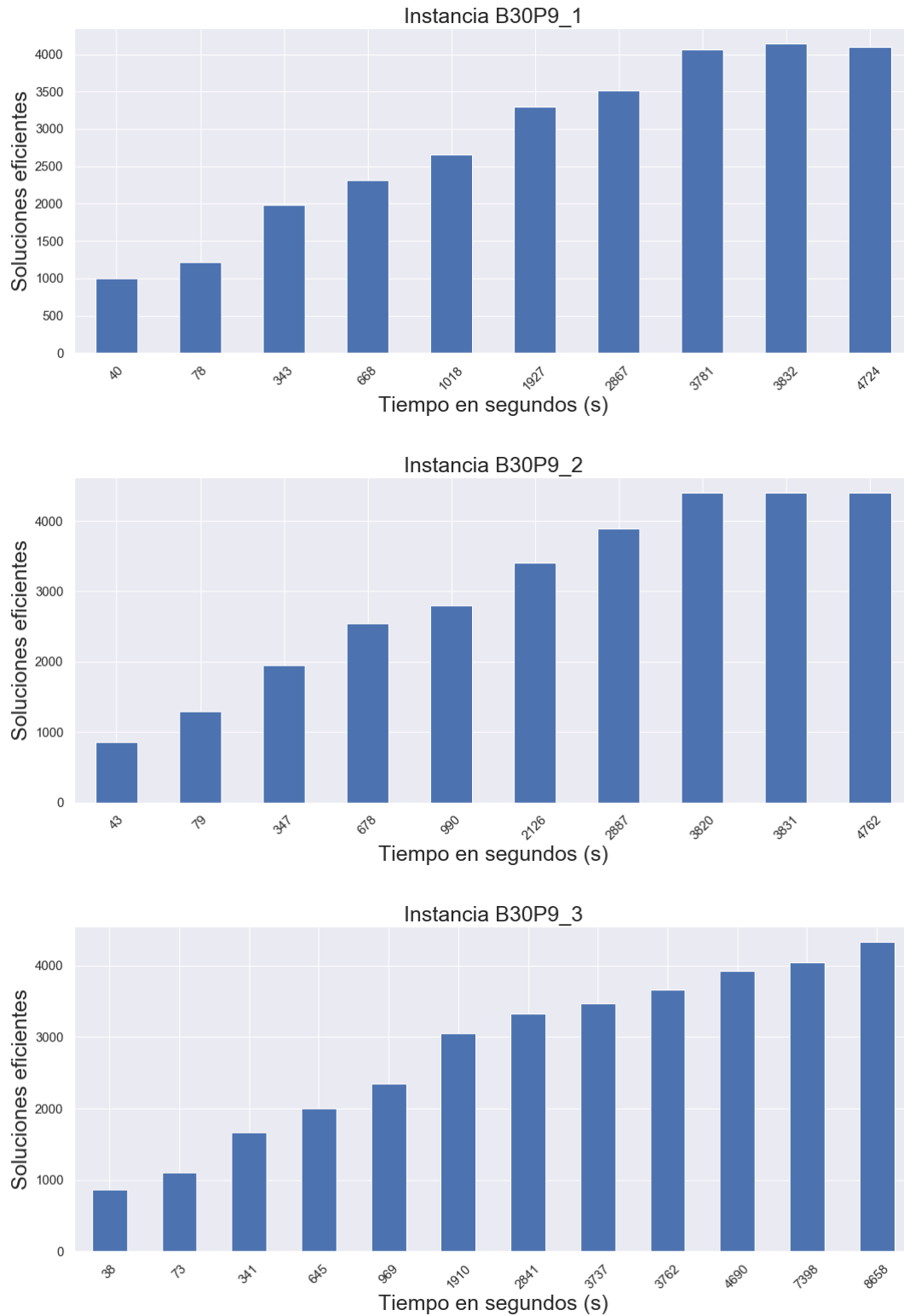
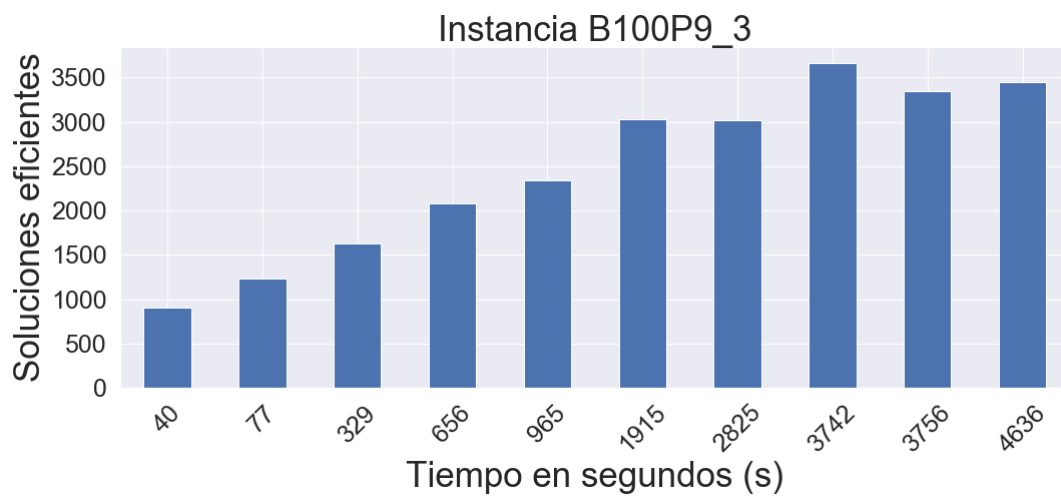
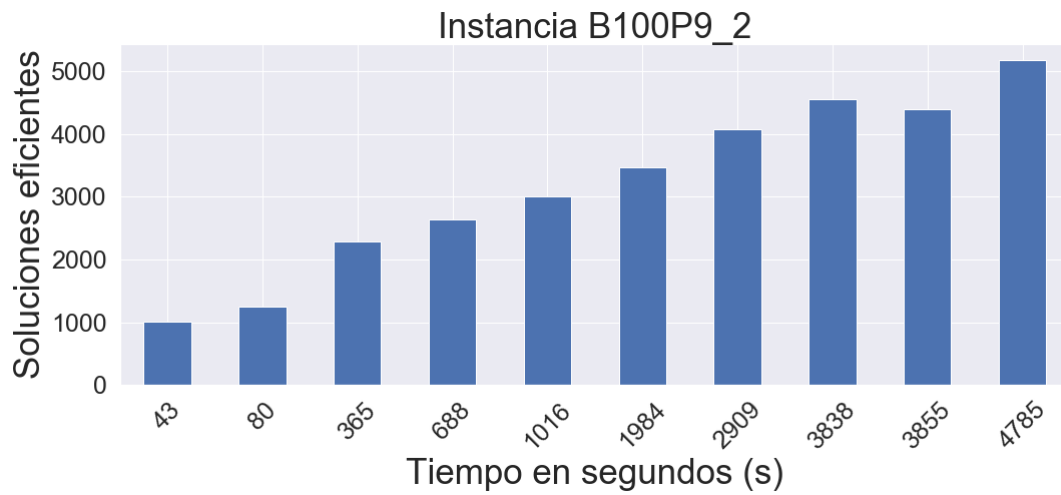
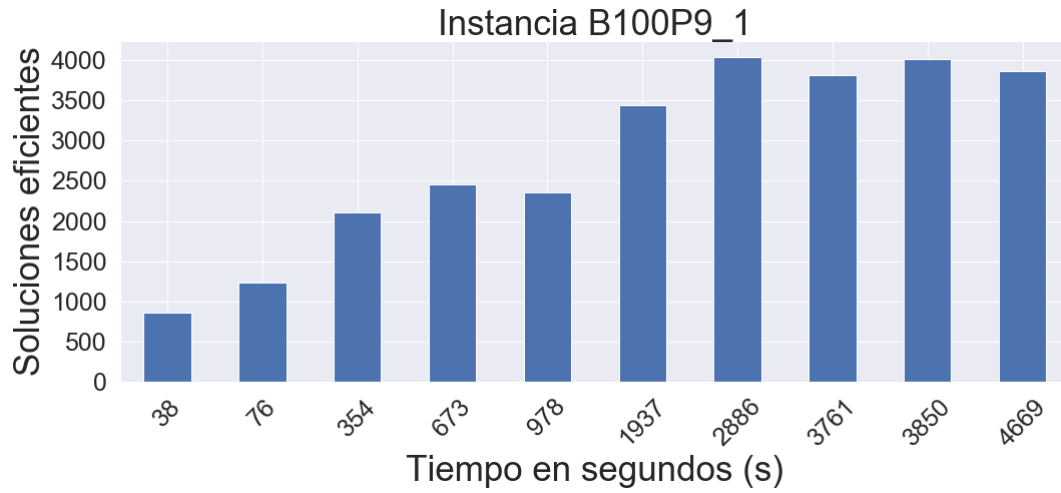


Figura 5.14: Cantidad de soluciones eficientes encontradas en instancias de 30 proyectos y nueve objetivos en el Grupo B.



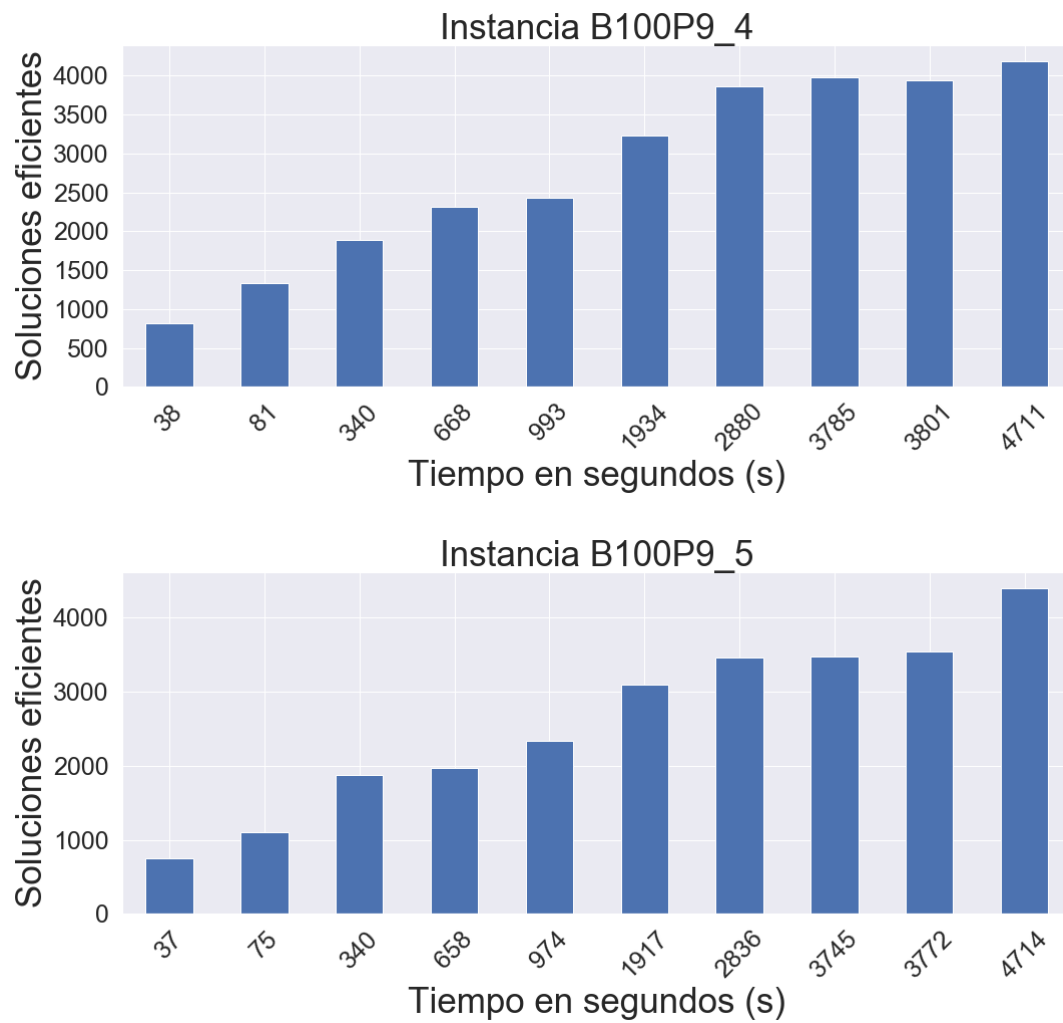


Figura 5.16: Cantidad de soluciones eficientes encontradas en instancias de 100 proyectos y nueve objetivos en el Grupo B.

Cuadro 5.4: Características de las instancias generadas Grupo C.

CI	P	O	A	R	AT	AP	T	S
4	64	4 a 6	3 a 5	2 a 4	No	Sí	No	No
4	64	4 a 6	3 a 5	2 a 4	No	Sí	No	2 a 8
4	128	4 a 6	3 a 5	2 a 4	No	Sí	No	No
4	128	4 a 6	3 a 5	2 a 4	No	Sí	No	8 a 12
4	256	4 a 6	3 a 5	2 a 4	No	Sí	No	No
4	256	4 a 6	3 a 5	2 a 4	No	Sí	No	14 a 38
4	512	4 a 6	3 a 5	2 a 4	No	Sí	No	No
4	512	4 a 6	3 a 5	2 a 4	No	Sí	No	12 a 42

tancias poseen cuatro objetivos, en el caso del grupo A, el tener solo dos objetivos y uno de ellos ser el cantidad de proyectos (número natural), hace que la cantidad de soluciones eficientes encontradas no sea relevante.

Las instancias son creadas originalmente con sinergias. Al momento de resolverlas estas se toman en cuenta o no, por lo que las instancias son las mismas, lo único que cambia es la inclusión de las sinergias. Las sinergias afectan directamente el beneficio, por lo que este tampoco resulta un parámetro de comparación.

El promedio de soluciones eficientes encontradas en diferentes tiempos se grafican la figura 5.17 para las instancias de 64 y 128 proyectos y en la figura 5.18 para las instancias de 256 y 512 proyectos. Para establecer una relación estadística entre la existencia o no de sinergias y la cantidad de soluciones eficientes encontradas en el tiempo, se realiza una prueba de hipótesis ANOVA [69] en estos grupos de instancias.

Variable independiente: Existencia o no de sinergias

Variable dependiente: Cantidad de soluciones no dominadas

Hipótesis nula H_0 : No existe relación entre las variables

Hipótesis alternativa H_1 : Existe relación entre las variables

Para las instancias de 64 proyectos el valor p obtenido es de 0.0093, hay menos de 1 % de probabilidades de que la hipótesis nula sea correcta por lo que se acepta la hipótesis alternativa. La cantidad de proyectos encontrados está relacionada con la con la existencia o no de sinergias.

Para las instancias de 128 proyectos, el valor p obtenido es de 0.0411. La evidencia en contra de la hipótesis nula es suficiente para rechazarla, por lo que se acepta la hipótesis alternativa.

En las instancias de 256 proyectos, el valor p obtenido es de 0.04708, con lo que se rechaza la hipótesis nula y se confirma la influencia de la existencia de sinergias en la cantidad de proyectos encontrados por el algoritmo. Igualmente ocurre en las instancias de 512 proyectos, en las que el valor p obtenido es de 0.01591.

Teniendo lo anterior en cuenta, se acepta la hipótesis de que la existencia de sinergias influye en el desempeño del algoritmo, particularmente en la cantidad de soluciones eficientes que puede encontrar en un tiempo dado.

Las heurísticas empleadas para agitar en este grupo fueron siete: `SwapRandom`, `SwapQuarter`, `SwapThird`, `SwapHalf`, `ShakeArea`, `ShakeRegion` y `AllRand`. Como puede apreciarse en la figura 5.19, al igual que en los grupos anteriores, `SwapQuarter` fue la que se empleó con más frecuencia.

Para la búsqueda local se emplearon igualmente siete heurísticas: `Swap1`, `DrawHightBgt`, `UseBgt`, `MoreProj`, `DrawRand`, `DrawRPutLess` y `SetPrjFill`. Como se observa en la figura 5.20, las que mejores rendimientos tuvieron fueron `DrawRPutLess`, `SetPrjFill` y `DrawRand`.

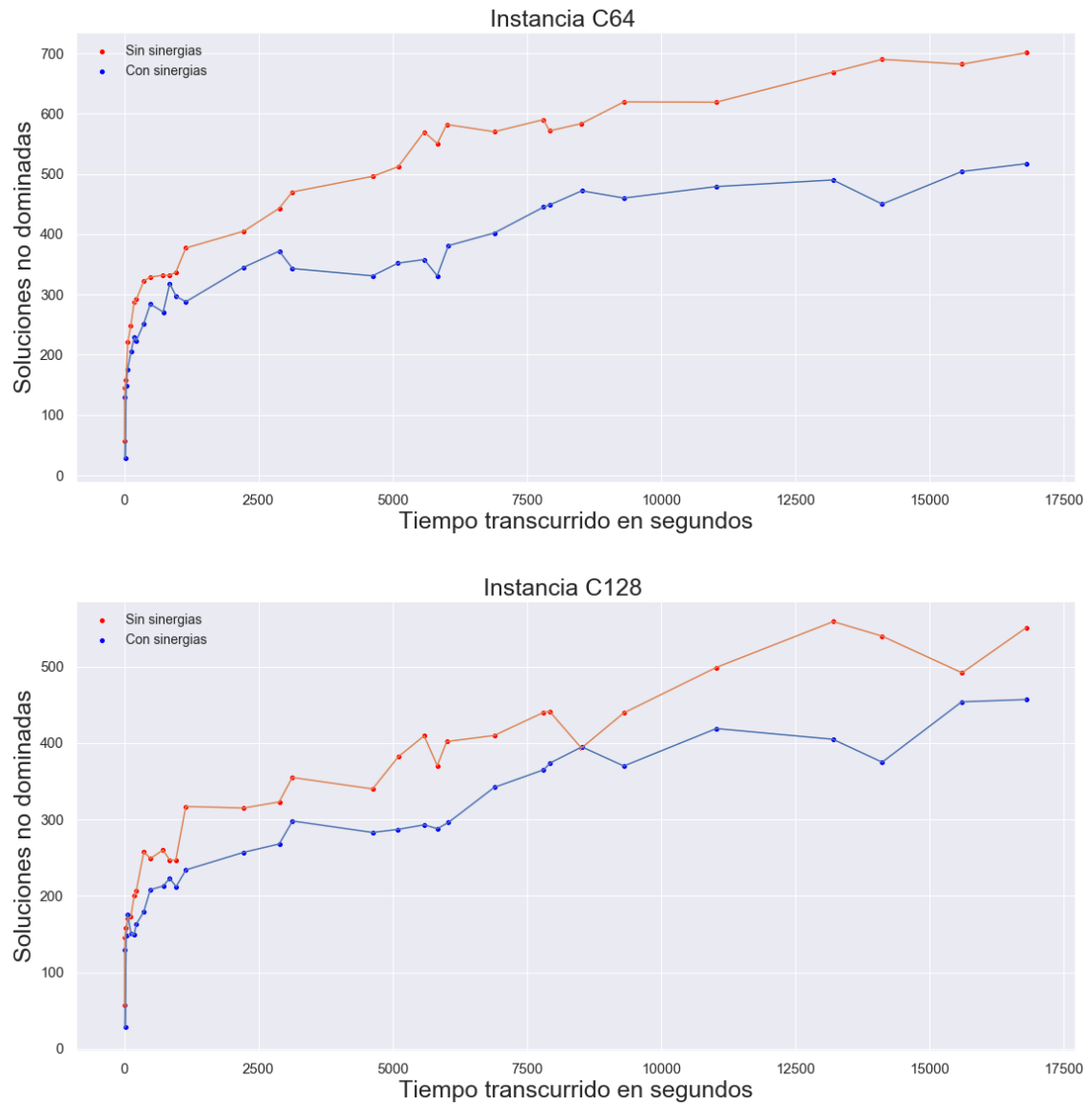


Figura 5.17: Cantidad de soluciones eficientes encontradas en las instancias de 64 y 128 proyectos en el Grupo C, con y sin sinergias.

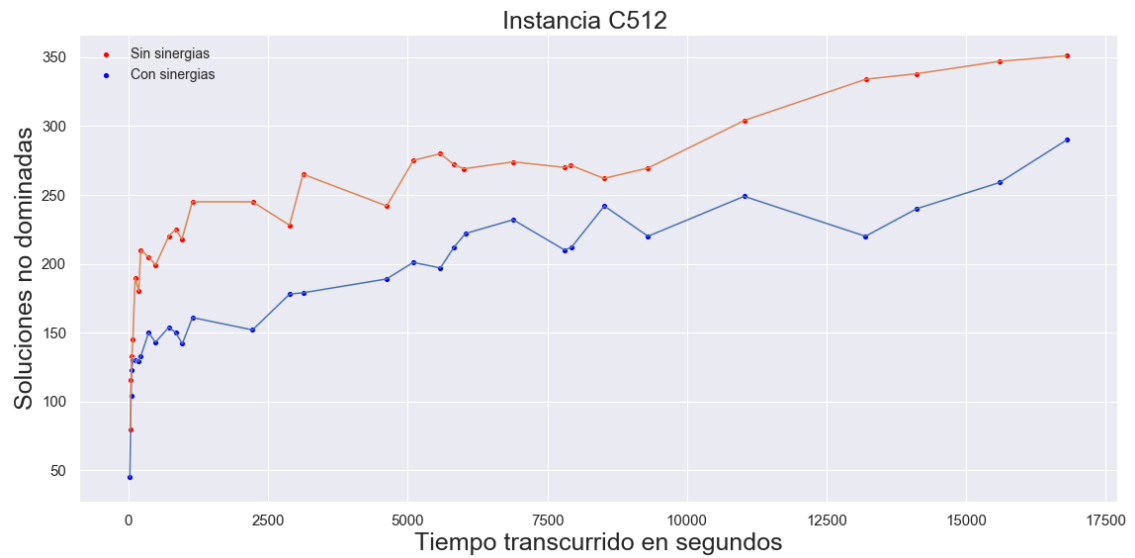


Figura 5.18: Cantidad de soluciones eficientes encontradas en las instancias de 256 y 512 objetivos en el Grupo C, con y sin sinergias.

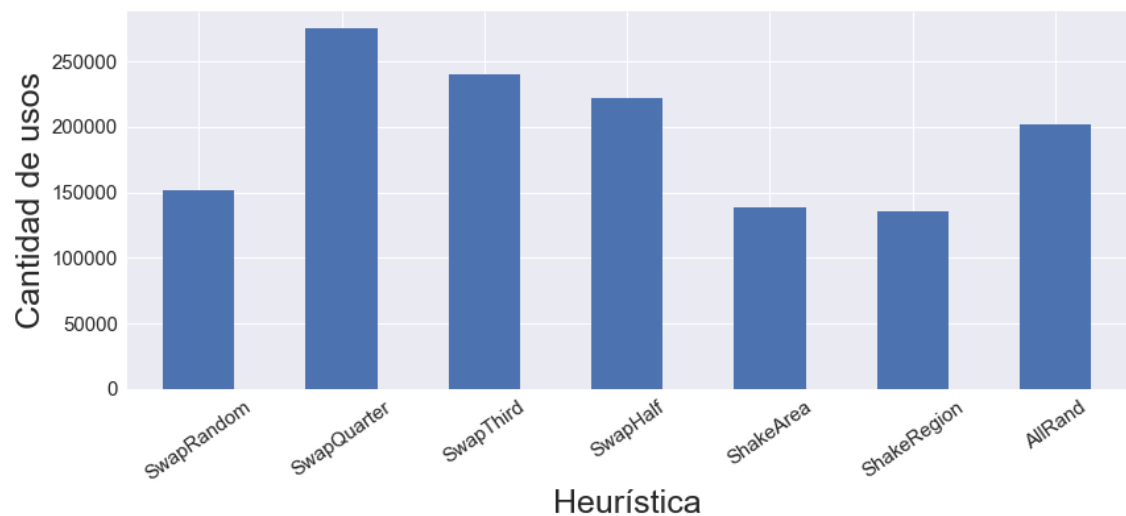


Figura 5.19: Desempeño de heurísticas de agitar para el Grupo C.

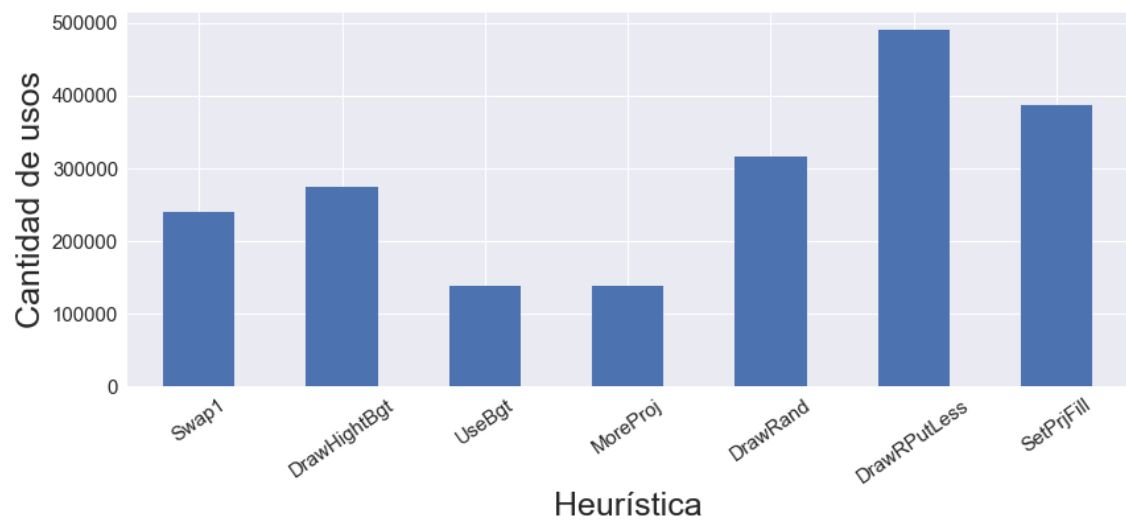


Figura 5.20: Desempeño de heurísticas de búsqueda local para el Grupo C.

CAPÍTULO 6

CONCLUSIONES

En la presente investigación se han analizado las principales características del problema de selección de cartera de proyectos con el fin de ofrecer soluciones a problemas con características diversas con un único algoritmo.

Se desarrolló el algoritmo HH-VNS-PP que ofrece soluciones a problemas de selección de cartera de proyectos estáticos, multiobjetivo y multicriterio. Estos problemas pueden tener asignación total o parcial y estar organizados en regiones, áreas y otros niveles organizacionales (tantos como se desee). Permite además granularidad de proyectos y tareas, con sinergias entre ellos. Se propone una metodología para el empleo de HH-VNS-PP como herramienta de apoyo a la toma de decisiones multicriterio.

HH-VNS-PP encontró soluciones localmente eficientes en las 140 instancias de prueba. La experimentación indica que el número de soluciones eficientes encontradas por el algoritmo disminuye con el número de proyectos y la presencia de sinergias. Teniendo esto en cuenta, como el algoritmo necesita, además, pocos recursos computacionales, lo hace ideal para organizaciones pequeñas y medias.

CAPÍTULO 7

TRABAJO FUTURO

Como trabajo futuro se pretende desarrollar experimentos computacionales en los que se obtengan soluciones exactas a problemas con diferentes características, empleando los mismos parámetros, para realizar un análisis profundo de la calidad de las soluciones ofrecidas por HH-VNS-PP.

Se considera conveniente incluir un mecanismo de memoria que le permita al algoritmo determinar cuál heurística es más eficiente aplicada inmediatamente luego de otra. Se propone, además, desarrollar experimentos para para ajustar los parámetros del algoritmo automáticamente en dependencia de las características de la instancia.

BIBLIOGRAFÍA

- [1] AARTS, E. y J. K. LENSTRA (2003), *Local search in combinatorial optimization*, Princeton University Press, isbn: 0-691-11522-2.
- [2] ARRATIA, N. (2017), *Metodología de apoyo a la decisión en la selección y planificación de carteras de proyectos de investigación y desarrollo bajo incertidumbre en organizaciones del sector público*, Tesis Doctoral, URL <https://hdl.handle.net/10630/16715>.
- [3] ARRATIA, N., R. CABALLERO, I. LITVINCHEV y F. LÓPEZ (2018), «Research and development project portfolio selection under uncertainty», *Journal of Ambient Intelligence and Humanized Computing*, **9**(3), págs. 857–866, doi: 10.1007/s12652-017-0564-7.
- [4] ARRATIA, N., S. E. SCHAEFFER, F. LÓPEZ y L. CRUZ (2016), «Static R&D project portfolio selection in public organizations», *Decision support systems*, **84**, págs. 53–63, doi: 10.1016/j.dss.2016.01.006.
- [5] ÁVILA TORRES, P. A. (2017), *Planificación multiperiodo de las frecuencias de paso y las tablas de tiempo con incertidumbre en la demanda y tiempo de viaje*, Tesis Doctoral, Universidad Autónoma de Nuevo León, URL https://riuma.uma.es/xmlui/bitstream/handle/10630/16760/TD_AVILA_TORRES_Paulina_Alejandra.pdf?sequence=1&isAllowed=n.
- [6] BARBERIS, G. F. y M. D. C. E. RÓDENAS (2011), «La Ayuda a la Decisión Multicriterio: orígenes, evolución y situación actual», en *VI Congreso Inter-*

- nacional de Historia de la Estadística y de la Probabilidad*, UNED, Universidad Nacional de Educación a Distancia, pág. 6, URL http://www.ahepe.es/VICongreso/descargas/Gabriela_Fdz_Barberis.pdf, isbn: 9788436264074.
- [7] BASTIANI, S., L. CRUZ, E. FERNÁNDEZ, C. GÓMEZ y V. RUIZ (2013), «Project ranking-based portfolio selection using evolutionary multiobjective optimization of a vector proxy impact measure», en *Fourth International Workshop on Knowledge Discovery, Knowledge Management and Decision Support*, Atlantis Press, doi: 10.2991/.2013.28.
- [8] BAZARAA, M. S., J. J. JARVIS y H. D. SHERALI (2011), *Linear programming and network flows*, John Wiley & Sons, isbn: 978-0-470-46272-0.
- [9] BHATTACHARYYA, R., P. KUMAR y S. KAR (2011), «Fuzzy R&D portfolio selection of interdependent projects», *Computers & Mathematics with Applications*, **62**(10), págs. 3857–3870, doi: 10.1016/j.camwa.2011.09.036.
- [10] BLUM, C. y A. ROLI (2003), «Metaheuristics in combinatorial optimization: Overview and conceptual comparison», *ACM Computing Surveys*, **35**(3), págs. 268–308, doi: 10.1145/937503.937505.
- [11] BRANS, J.-P., P. VINCKE y B. MARESCHAL (1986), «How to select and how to rank projects: The PROMETHEE method», *European journal of operational research*, **24**(2), págs. 228–238, doi: 10.1016/0377-2217(86)90044-5.
- [12] BURKE, E. K., M. R. HYDE, G. KENDALL, G. OCHOA, E. ÖZCAN y J. R. WOODWARD (2019), «A classification of hyper-heuristic approaches Revisited», en *Handbook of Metaheuristics*, Springer, págs. 453–477, doi: 10.1007/978-1-4419-1665-5, isbn: 978-1-4419-1665-5.
- [13] BURKE, E. K. y G. KENDALL (2005), *Search methodologies*, Springer, doi: 10.1007/978-1-4614-6940-7.
- [14] CARAZO, A. F., T. GÓMEZ, J. MOLINA, A. G. HERNÁNDEZ-DÍAZ, F. M. GUERRERO y R. CABALLERO (2010), «Solving a comprehensive model for mul-

- tiobjective project portfolio selection», *Computers & operations research*, **37**(4), págs. 630–639, doi: 10.1016/j.cor.2009.06.012.
- [15] CARAZO, A. F., T. G. NÚÑEZ y F. PÉREZ (2011), «Análisis de los Principales Aspectos que Afectan a la Decisión de Selección y Planificación de Cartera de Proyectos», *Rect@: Revista Electrónica de Comunicaciones y Trabajos de ASEPUMA*, (12), págs. 123–140, URL https://www.researchgate.net/publication/285206328_Analisis_de_los_principales_aspectos_que_afectan_a_la_decision_de_seleccion_y_planificacion_de_cartera_de_proyectos.
- [16] CHAKHLEVITCH, K. y P. COWLING (2008), «Hyperheuristics recent developments», en *Adaptive and multilevel metaheuristics*, Springer, págs. 3–29, doi: 10.1007/978-3-540-79438-7.
- [17] CHÁVEZ, M. A. C. (2018), «Un Mecanismo de Vecindad con Búsqueda Local y Algoritmo Genético para el Problema de Transporte con Ventanas de Tiempo», *Programación Matemática y Software*, URL <http://riaa.uaem.mx/handle/20.500.12055/63>.
- [18] CONTI, D., C. SIMÓ y A. RODRÍGUEZ (2005), «Teoría de carteras de inversión para la diversificación del riesgo enfoque clásico y uso de redes neuronales artificiales (RNA)», *Ciencia e Ingeniería*, **26**(1), págs. 35–42, URL <http://erevistas.saber.ula.ve/index.php/cienciaeingenieria/article/view/318>.
- [19] COOPER, W. W., L. M. SEIFORD y K. TONE (2006), *Introduction to data envelopment analysis and its uses: with DEA-solver software and references*, Springer Science & Business Media, doi: 10.1007/0-387-29122-9.
- [20] COWLING, P., G. KENDALL y E. SOUBEIGA (2000), «A hyperheuristic approach to scheduling a sales summit», en *International Conference on the Practice and Theory of Automated Timetabling*, Springer, págs. 176–190, doi: 10.1007/3-540-44629-X₁1.

-
- [21] CRUZ, L., E. FERNANDEZ, C. GOMEZ, G. RIVERA y F. PEREZ (2014), «Many-objective portfolio optimization of interdependent projects with'a priori'incorporation of decision-maker preferences», *Applied Mathematics & Information Sciences*, **8**(4), pág. 1517, doi: 10.12785/amis/080405.
- [22] DAENGDEJ, J., D. LUKOSE y R. MURISON (1999), «Using statistical models and case-based reasoning in claims prediction: experience from a real-world problem», en *Applications and Innovations in Expert Systems VI*, Springer, págs. 217–229, doi: 10.1016/S0950-7051(99)00015-5.
- [23] DE LOS ÁNGELES, M. (2015), *Apoyo a la decisión para el diseño y la planeación integrados de una cadena de suministro*, Tesis Doctoral, Universidad Autónoma de Nuevo León, URL <http://eprints.uanl.mx/9257>.
- [24] DENZINGER, J. y M. FUCHS (1996), «High performance ATP systems by combining several AI methods», *SEKI Report*, (96,9), URL <https://www.ijcai.org/Proceedings/97-1/Papers/017.pdf>.
- [25] DÍAZ, A. y F. GLOVER (2000), «Optimización heurística y redes neuronales», *Paraninfo*, isbn: 9788428322690.
- [26] DOERNER, K., W. J. GUTJAHR, R. F. HARTL, C. STRAUSS y C. STUMMER (2004), «Pareto ant colony optimization: A metaheuristic approach to multiobjective portfolio selection», *Annals of operations research*, **131**(1-4), págs. 79–99, doi: 10.1023/B:ANOR.0000039513.99038.c6.
- [27] DYE, L. D. y J. S. PENNYPACKER (1999), *Project portfolio management selecting and prioritizing projects for competitive advantage*, Center for Business Practices, isbn: 1929576005.
- [28] EILAT, H., B. GOLANY y A. SHTUB (2006), «Constructing and evaluating balanced portfolios of R&D projects with interactions: A DEA based methodology», *European journal of operational research*, **172**(3), págs. 1018–1039, doi: 10.1016/j.ejor.2004.12.001.

-
- [29] FERNANDEZ, E., F. LOPEZ, J. NAVARRO, I. VEGA y I. LITVINCHEV (2009), «An integrated mathematical-computer approach for R&D project selection in large public organisations», *International Journal of Mathematics in Operational Research*, **1**(3), págs. 372–396, doi: 10.1504/IJMOR.2009.024291.
- [30] FERNANDEZ, E. y R. OLMEDO (2013), «Public project portfolio optimization under a participatory paradigm», *Applied computational intelligence and soft computing*, **2013**, pág. 4, doi: 10.1155/2013/891781.
- [31] GALVÁN, F. (2017), *Híperheurístico basado en Búsqueda Local Iterativa para la solución de un problema de cartera de proyectos*, Tesis de Maestría, Universidad Autónoma de Nuevo León, URL <http://eprints.uanl.mx/16552/1/1080289815.pdf>.
- [32] GARCÍA RODRÍGUEZ, R. (2010), *Hiper-heurístico para Resolver el Problema de Cartera de Proyectos Sociales*, Tesis de Maestría, Instituto Tecnológico de Ciudad Madero, URL <http://eprints.uanl.mx/16552/1/1080289815.pdf>.
- [33] GHAPANCHI, A. H., M. TAVANA, M. H. KHAKBAZ y G. LOW (2012), «A methodology for selecting portfolios of projects with interactions and under uncertainty», *International Journal of Project Management*, **30**(7), págs. 791–803, doi: 10.1016/j.ijproman.2012.01.012.
- [34] GHASEMZADEH, F. y N. P. ARCHER (2000), «Project portfolio selection through decision support», *Decision support systems*, **29**(1), págs. 73–88, doi: 10.1016/S0167-9236(00)00065-8.
- [35] GHOSH, A., P. MAL y A. MAJUMDAR (2019), *Elimination and Choice Translating Reality (ELECTRE)*, págs. 65–84, doi: 10.1201/9780429504419-4.
- [36] GUTIÉRREZ-URZÚA, M. I., P. GALVEZ-GALVEZ, B. ELTIT y H. REINOSO (2017), «Resolución del problema de carteras de inversión utilizando la heurística de colonia artificial de abejas», *Estudios gerenciales*, **33**(145), págs. 391–399, doi: 10.1016/j.estger.2017.11.001.

- [37] HANSEN, P. y N. MLADENović (1997), «Variable neighborhood search for the p-median», *Location Science*, **5**(4), págs. 207–226, doi: 10.1016/S0966-8349(98)00030-8.
- [38] HANSEN, P. y N. MLADENović (2003), «Variable neighborhood search», en *Handbook of metaheuristics*, Springer, págs. 145–184, doi: 10.1007/978-1-4419-1665-5.
- [39] HANSEN, P., N. MLADENOVIC, J. BRIMBERG y J. A. M. PÉREZ (2019), «Variable Neighborhood Search», en *Handbook of metaheuristics*, Springer, págs. 60–86, doi: 10.1007/978-1-4419-1665-5.
- [40] HASSANZADEH, F., M. MODARRES, H. R. NEMATI y K. AMOAKO-GYAMPAH (2014), «A robust R&D project portfolio optimization model for pharmaceutical contract research organizations», *International Journal of Production Economics*, **158**, págs. 18–27, doi: 10.1016/j.ijpe.2014.07.001.
- [41] HSIAO, P.-C., T.-C. CHIANG y L.-C. FU (2012), «A vns-based hyper-heuristic with adaptive computational budget of local search», en *2012 IEEE Congress on Evolutionary Computation*, IEEE, págs. 1–8, doi: 10.1109/CEC.2012.6252969.
- [42] JUNGUITO, R., J. J. PERFETTI y A. BECERRA (2014), *Desarrollo de la agricultura colombiana*, URL https://repository.fedesarrollo.org.co/bitstream/handle/11445/151/CDF_No_48_Marzo_2014.pdf, isbn: 978-958-57963-9-3.
- [43] KALISZEWSKI, I. y D. PODKOPAEV (2016), «Simple additive weighting—A meta-model for multiple criteria decision analysis methods», *Expert Systems with Applications*, **54**, págs. 155–161, doi: 10.1016/j.eswa.2016.01.042.
- [44] KONIDARI, P. y D. MAVRAKIS (2007), «A multi-criteria evaluation method for climate change mitigation policy instruments», *Energy Policy*, **35**(12), págs. 6235–6257, doi: j.enpol.2007.07.007.
- [45] LITVINCHEV, F., I LÓPEZ, H. ESCALANTE y M. MATA (2011), «A milp bi-objective model for static portfolio selection of R&D projects with synergies», *Jour-*

- nal of Computer and Systems Sciences International*, **50**(6), págs. 942–952, doi: /10.1134/S1064230711060165.
- [46] LITVINCHEV, I., F. LÓPEZ, N. ARRATIA y J. A. MARMOLEJO (2014), «Selecting large portfolios of social projects in public organizations», *Mathematical Problems in Engineering*, **2014**, doi: 10.1155/2014/654293.
- [47] LITVINCHEV, I. S., F. LÓPEZ, A. ALVAREZ y E. FERNÁNDEZ (2010), «Large-scale public R&D portfolio selection by maximizing a biobjective impact measure», *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, **40**(3), págs. 572–582, doi: 10.1109/TSMCA.2010.2041228.
- [48] LOURENÇO, H. R., O. C. MARTIN y T. STÜTZLE (2019), «Iterated local search: Framework and applications», en *Handbook of metaheuristics*, Springer, págs. 129–168, doi: 10.1007/978-1-4419-1665-5.
- [49] MARTI, R. (2003), «Procedimientos metaheurísticos en optimización combinatoria», *Matemàtiques, Universitat de València*, **1**(1), págs. 3–62, URL <http://www.uv.es/rmarti/paper/docs/heur1.pdf>.
- [50] MURILLO, M. A. B., F. A. M. PEÑA y Ó. P. LEÓN (2017), «Optimización de portafolio de proyectos a través de la aplicación de programación lineal y el CAPM», *Revista Ciencias Estratégicas*, **25**(37), págs. 71–86, isbn: 9788436264074.
- [51] NI, Q., X. YIN, K. TIAN y Y. ZHAI (2017), «Particle swarm optimization with dynamic random population topology strategies for a generalized portfolio selection problem», *Natural Computing*, **16**(1), págs. 31–44, doi: 10.1007/s11047-016-9541-x.
- [52] OSMAN, J. P., IBRAHIM H & KELLY (1997), «Meta-heuristics theory and applications», *Journal of the Operational Research Society*, **48**(6), págs. 657–657, doi: 10.1007/978-1-4613-1361-8, isbn: 978-1-4612-8587-8.
- [53] ÖZCAN, E., B. BILGIN y E. E. KORKMAZ (2008), «A comprehensive analysis of hyper-heuristics», *Intelligent Data Analysis*, **12**(1), págs. 3–23, doi: 10.3233/IDA-2008-12102.

-
- [54] PACHECO, J. y C. DELGADO (2000), «Resultados de Diferentes Experiencias con Búsqueda Local Aplicadas a Problemas de Rutas», *Revista Electrónica Rect@. ASE-PUMA*, **2**(1), págs. 54–81, URL https://www.researchgate.net/publication/26423606_Resultados_de_diferentes_experiencias_con_busqueda_local_aplicadas_a_problemas_de_rutas.
- [55] PAYNE, J. W. y E. J. BETTMAN, JAMES R & JOHNSON (1993), *The adaptive decision maker*, Cambridge University Press, Australia, isbn: 0-521-41505-5.
- [56] PÉREZ, F., T. GÓMEZ, R. CABALLERO y V. LIERN (2018), «Project portfolio selection and planning with fuzzy constraints», *Technological Forecasting and Social Change*, **131**, págs. 117–129, doi: 10.1016/j.cor.2009.06.012.
- [57] RISAWANDI, R. R. (2016), «Study of the simple multi-attribute rating technique for decision support», *Decision-making*, **4**, pág. C4, doi: 10.1088/1742-6596/930/1/012015.
- [58] ROMERO, C. (1996), *Análisis de las decisiones multicriterio*, tomo 14, Isdefe Madrid, isbn: 84-89338-14-0.
- [59] ROMERO, C. (2014), *Handbook of critical issues in goal programming*, Elsevier, isbn: 9781483295114.
- [60] ROMMERT STELLINGWERF, A. Z. (2013), *ISO 21500 Guidance on project management – A Pocket Guide*, Van Haren Publishing, Zaltbommel, URL https://www.vanharen.net/amfilerating/file/download/file_id/130/.
- [61] ROY, B. (2013), *Multicriteria methodology for decision aiding*, tomo 12, Springer Science & Business Media, isbn: 079234166X.
- [62] SAATY, T. L. (2004), «Decision making—the analytic hierarchy and network processes (AHP/ANP)», *Journal of systems science and systems engineering*, **13**(1), págs. 1–35, doi: 10.1007/s11518-006-0151-5.
- [63] SCHRIJVER, A. (1998), *Theory of linear and integer programming*, John Wiley & Sons, isbn: 978-0-470-46272-0.

- [64] SIMON, H. A. (1960), «The new science of management decision», isbn: 0136161367.
- [65] SIXTO RÍOS, A. M. Y. A. J. (2004), «Optimización multiobjetivo basada en metaheurísticas», *Anales de la Real Academia de Doctores de España*, **8**, págs. 159–177, URL https://www.researchgate.net/publication/236168899_Optimizacion_Multiobjetivo_basada_en_Metaheuristicas.
- [66] SOLAK, S., J.-P. B. CLARKE, E. L. JOHNSON y E. R. BARNES (2010), «Optimization of R&D project portfolios under endogenous uncertainty», *European Journal of Operational Research*, **207**(1), págs. 420–433, doi: 10.1016/j.ejor.2010.04.032.
- [67] STUMMER, C. y K. HEIDENBERGER (2003), «Interactive R&D portfolio analysis with project interdependencies and time profiles of multiple objectives», *IEEE Transactions on Engineering Management*, **50**(2), págs. 175–183, doi: 10.1109/TEM.2003.810819.
- [68] SUÁREZ, L. M. B., N. R. O. PIMIENTO y J. B. D. DUARTE (2015), «Selección de portafolios de inversión socialmente responsables usando el método de las restricciones y la técnica multicriterio proceso analítico jerárquico», *Revista EIA*, **12**(24), págs. 71–85, doi: 10.24050/reia.v0i0.634.
- [69] TABACHNICK, B. G. y L. S. FIDELL (2007), *Experimental designs using ANOVA*, Thomson/Brooks/Cole Belmont, CA, isbn: 0534405142.
- [70] TZENG, G.-H. y J.-J. HUANG (2011), *Multiple attribute decision making: methods and applications*, CRC press, URL <https://dialnet.unirioja.es/descarga/articulo/4215171.pdf>.
- [71] VELASQUEZ, M. y P. T. HESTER (2013), «An analysis of multi-criteria decision making methods», *International journal of operations research*, **10**(2), págs. 56–66, URL https://www.researchgate.net/publication/275960103_An_analysis_of_multi-criteria_decision_making_methods.

-
- [72] WOLSEY, L. A. (1998), *Integer programming*, John Wiley & Sons, isbn: 9781119606536.
- [73] ZADEH, L. A. (1965), «Fuzzy sets», *Information and control*, **8**(3), págs. 338–353, doi: 10.1016/S0019-9958(65)90241-X.
- [74] ZÁRATE, G. R. (2015), «Solución a Gran Escala Del Problema de Cartera de Proyectos Caracterizados con Múltiples Criterios», *Instituto Tecnológico de Tijuana*, doi: 10.13140/2.1.2798.4160.

RESUMEN AUTOBIOGRÁFICO

Maday Hernández Quevedo

Candidato para obtener el grado de
Maestría en Ciencias de la Ingeniería con Orientación en Sistemas

Universidad Autónoma de Nuevo León
Facultad de Ingeniería Mecánica y Eléctrica

Tesis:

HIPERHEURÍSTICO MULTIOBJETIVO BASADO EN BÚSQUEDA DE
VECINDAD VARIABLE PARA PROBLEMAS DE SELECCIÓN DE
CARTERA DE PROYECTOS

Nacida en La Habana, Cuba, el 10 de junio de 1988. Graduada de la Facultad de Ingeniería Eléctrica de la Universidad de Oriente como Ingeniera en Telecomunicaciones y Electrónica. Trabajé como Técnico en Informática en FATESA (2007-2010), como ingeniera en el Instituto de Medicina Tropical Pedro Kouri y como Ingeniera Especialista a cargo de las Radiocomunicaciones en la Dirección Nacional de Comunicaciones del MINSAP (2018).